CAPTIVA

SOFTWARE CORPORATION

FormWare File System Guide

Version 5.0

Contents

2   Installing & Configuring 2-1

1 *Getting Started*

Overview

Thank you for choosing FormWare™ for your forms processing needs.

Automating the data entry process can be a complex, even difficult, task. Before you can develop an efficient and cost-effective forms processing application, you must understand the data capture process. It is also important to understand good form design. A well-designed form facilitates automated data entry and improves the accuracy of recognition technologies.

The FormWare forms processing software is a collection of programs designed specifically for the Windows environments and is modular in structure. This means that to design and run your forms processing application, you can purchase the specific modules you need.

The Understanding FormWare Modules section gives a high-level explanation of each module to present key features and functionality, and show how the modules fit together. By understanding the modules, you can design a forms processing application tailored specifically to the needs of your organization.

This guide is designed to present FormWare users with conceptual information about the FormWare product as well as step-by-step instructions on how to use the software. For the latest information, be sure to check any Release Notes and Read Me files included with the software.

The following sections are included in this guide:

Getting Started – Provides high-level conceptual information on FormWare. Including:

- Understanding Data Capture and Forms Processing
- Identifying the Steps in a JobFlow
- Understanding Components
- System Administrator Responsibilities
- Analyzing Form Design

Installing & Configuring – Provides detailed instructions for installing and configuring FormWare to your particular environment.

Designing – Provides detailed instructions on how to design a FormWare application from start to finish.

Testing – Provides instructions on how to test the different components of your FormWare application prior to moving it to production.

Using – Provides instructions on some of the common tasks performed with a FormWare application in a production environment.

Maintaining – Provides instructions on some of the common maintenance procedures you can perform on your FormWare application.

Troubleshooting – Contains helpful tips and tricks on how to solve common problems with the different FormWare components.

Supplementary Information – Contains reference information on:

- INI Parameters
- Client Parameters
- Form ID Training Parameters

Understanding Data Capture & Forms Processing

The term 'forms processing' means different things to different people. For one organization, forms processing might involve operators typing all data from paper or from images into fields on an electronic form. Another organization may use recognition technologies to capture the data and then use operators only to fix questionable characters and perform data verification . While the FormWare software can be used in each of these scenarios, it is primarily designed for use in forms processing applications that use recognition technologies.

In a Key-from-Paper (KFP) application, the operator manually types information into the data form from the hard copy document rather than from an electronic image. You can design a KFP application by setting up a shortcut for the completion operator to access the job and batch to be keyed. When the operator finishes keying, the batch is added to the appropriate JobFlow task.

If a document gets rejected with an image quality problem, a KFP or Key-from-Image (KFI) process can be used. In a KFI application, operators manually identify the images, type the correct form numbers during Form Identification, and type data into the data form during Completion. You can also design a KFI application by setting it to Image Only.

Identifying the Steps in a JobFlow

Document Preparation	The first step in the data capture process is to prepare paper documents for the scanning process. This step can involve:
	• Opening envelopes and unfolding the documents to ready them for the scan process.
	• Placing a stack of documents right side up in a page "jogger" to align all document edges and reduce the possibility of document misfeed during scanning.
	• Using batch tickets to divide groups of documents into separate batches.
	• Deriving a batch total and writing this on the batch ticket if checks or dollar amounts are part of the document stack.
	• Depending on the type of documents you are scanning, additional preparation steps may also be necessary.

Scanning	After the paper documents are ready, the next step is scanning the documents to produce electronic images. You should scan the documents using a method that produces the cleanest possible image. For optimal results, set the scanner to the lowest DPI that creates a readable image.
	Another important part of the scanning process is adjusting the contrast and intensity to improve the quality of images, reduce specks on images, and remove light, unwanted lines. You can also scan a portion of the document instead of the entire document, resulting in a smaller, less-cluttered image.
Image Pre-Processing	Batches added to a workflow are automatically routed to the first task in the system, usually an image processing or pre-processing task. Because no high-production scanner creates perfectly aligned images, it is a good idea to use image enhancement tools to de-skew and register images. Even the best scanners can produce images with a slight skew, which can reduce the recognition accuracy rate.
	In addition, each image needs a common point on which to base all recognition attributes. This is called a registration mark. If you do not register your image, the recognition accuracy rate can degrade.
	The image pre-processing functions of line removal, speck removal, character smoothing, and image inversion can also improve the recognition accuracy rate. If you scan documents at a 90°, 180°, or 270° rotation due to an edge perforation, you can use the rotation function to rotate the image to the proper orientation.
Image Quality	After processing images, operators can visually verify image quality before the image is processed any further. The image quality check process identifies image problems such as a dark image, light contrast, holes, torn pages, or bent corners. After identifying a problem, the operator can route the image to a queue for rescanning or other correction.
Form Identification	If you have different form types in the same batch, typically the next step is form identification (Form ID). The Form ID process uses various methods to synchronize the image being processed with the correct Form Definition file (FDF). The FDF contains recognition attributes and validation checks and is unique for each form type. Without Form ID, document preparation would be much more labor-intensive, and this extra time and effort could increase costs.
	You can achieve Form ID using a physical attribute on the image, such as a bar code or a document identifier number or based on the file size of the image or its height and width in pixels.
Image Enhancement	Image enhancement further improves pre-processed images to prepare them for recognition. It reduces the overall image size by removing unnecessary information, such as extraneous text, company logos, labels, boxes, or any information that is not required for Recognition.
	Image enhancement performs functions on the portion of the image defined within a recognition zone, allowing you to enhance an image at a more detailed level than the page level. The process results in an "enhanced image" that can either overwrite the original image or be output as a separate image to a different directory.
Recognition	Recognition converts the pixels of the text on the image into characters that are inserted into the fields of the data form. The type of data recognized on the image can be machine print, handprint, cursive, mark sense, or bar code. Once recognition is complete, you can use rules to run validation over the data.

Completion	During Completion, you can perform reject repair, data entry, and data validation on all images in a batch. By combining the speed of out-of-context reject repair with the accuracy of in-context repair, Completion offers a one-pass approach for reject repair that maximizes performance and operator productivity.
	You can also use a multiple-pass approach to perform reject repair on batch images. By separating fields on the form by type (e.g., all numeric or all alphabetic fields) or some other user-defined criteria, only specific fields display to the operator during a single reject repair pass. For example, in a typical multiple-pass approach, one pass would repair only numeric fields, another pass only alphabetic fields, a third alpha-numeric fields, and a final pass enter data for non-recognized fields. Separating reject repair passes by data type—numeric fields from alphabetic fields—allows operators to repair characters faster.
Verification	Verification is the only way to avoid costly errors in character substitutions and ensure high data accuracy. During verification, the operator retypes each character and the system compares it with the character generated by the recognition engine.
	You can decide to apply this type of blind verification on all fields or only selected fields. When the verification process finds a discrepancy, it displays a message that instructs the operator to correct or re-key data.
Post-Completion File Edits	During the post-completion file edit process, an edit program is run over the batch data files to check calculations, perform table validations, copy files from one directory to another, or perform table updates to further ensure batch data integrity. If the edit discovers no problems, the batch is ready for export and is routed to the export process. If a problem is found within a batch, the edit program can route the batch data file to an error queue for repair.
Post-Completion Repairs	The post-completion repair process corrects problems identified in the BDF. These problems can be flagged either by an operator during reject repair or by an edit program during the post-completion file edit process. During post-completion repair, only those fields and records that have problems display to the operator. The operator can then determine why the problem occurred and either repair or delete the record. Once this process is complete, the BDF returns to the post-completion edit task so the edit program can verify all changes. If the BDF is free of errors, it can then be routed to the data export step.
Exporting Data	The ultimate reason to process forms is to extract data from a form and export it into the format required by your database or image archiving system. Exporting any valid, completed batch data files can be performed either manually by a supervisor or set up to run as an automatic export task within the workflow. After batches are exported, the data is written to a DAT tape, a disk, or uploaded to a mainframe.
Clean Up	After batches are successfully exported, the Clean Up process removes all files associated with a batch—image files, batch data files, BDFs, and recognition results files—from the system. Since Clean Up deletes these files from the system, it is important that you perform this task only after you verify that the BDFs have been successfully exported and transmitted to the appropriate location. You can also choose to delete the folder that contains the batch images. By performing the Clean Up task, you can process large numbers of images without needing significant hard disk space.

When you design your workflow routing, you can route batches from process to process without manual intervention. For example, you can route a batch automatically from scanning to image pre-processing, to recognition, to completion, to verification, and then to export. Using the features of a workflow component minimizes repetitive, administrative tasks, which ultimately maximizes an operator's time and productivity.

Workflow routing can also assign batches to operators, prevent operators from verifying their own work, group batches of work together, and assess batch priority numbers, providing seamless integration with third-party products and custom programs. An open Application Program Interface (API) can allow for the creation of custom edits for use on all batches that are part of the workflow system.

Understanding Components

The FormWare product is comprised of several components. When you purchase FormWare, you decide which of the following components you will need within your system.

Administration

The Administration component provides the necessary maintenance functions to manage a high-volume, forms processing application. When you create your forms processing application, you generate a Form Definition File (FDF) as well as other components necessary to run the application. During file creation and editing, the Administration component manages the storage and maintenance of these files. The Administration component enables you to define additional settings (e.g., check digit algorithms or value tables) as well as maintain files and images.

The Administration component has the following features:

- **Browser Interface** - The Administration component displays all FormWare files and folders using a Microsoft Explorer-like browser. Because this browser interface is familiar to Windows users, they can easily manage the images, files, and other components of the FormWare system. Each function in the system has its own folder containing all files for that particular function.

- **Virtual Directories** - Many image-based forms processing systems segregate images and Recognition results into subdirectories, which are usually based on a customer name or Julian date. This means that a job can have associated images and data in many different locations on the system. The virtual directories feature allows you to select one folder, such as the JOB FILES folder, to display all data files in all directories for a specific job. With virtual directories, you can maintain specific files, including image files, for your application.

- **System Security Using INI Files** - Within the Administration component, you can secure access to the various modes of the Completion component using special user or group Initialization (INI) files. You use INI files to customize user, job, and system settings. The INI files within the Administration component control the users who can access files for the Reject Repair, Verification, or Remove Flags modes of the Completion component. The INI files also control access to the administration functions. By creating user INI files, you can grant limited rights to individual users.

Designer

The Designer component provides a development environment for designing and testing your forms processing application. From Designer's graphical user interface, you can access application design tools, which enable you to create and modify elements of a data form or an I-Form with the click of a button or the drag of a mouse. For example, you can enable validation edits such as right-justifying numeric fields, filling fields with spaces when empty, and performing range checks to determine if a number falls within a specified range.

You can develop your applications and test them repeatedly until you achieve the optimum combination of forms processing functions and properties. To test the accuracy of each process, you can run all processing components from the Designer component. You can also choose to test a single process on the form and field levels, or you can test a number of processes simultaneously on the job, form, and field levels.

In addition to the default edits provided with the software, you can create unique Dynamic Link Libraries (DLLs) or write custom edit programs to tailor the software to specific form types. The Designer component also provides access to Visual Basic for Applications (VBA) technology, allowing you to leverage the functionality of your forms processing application by designing special processing programs that link into that application.

Scan

Scanning Capabilities

The Scan module provides advanced scanning capabilities for successful batch-based forms processing applications. The Scan module is designed for use with all of the components of the FormWare software, creating Group4 TIFF images and BDFs needed for a seamless forms processing system.

You can use the Scan module to scan a single paper form, which allows you to test scanner parameters such as contrast, intensity, and dots per inch. The Scan module supports single or duplex tray scanning, portrait or landscape orientation, and most common paper sizes. Support for these options, as well as for contrast and intensity, depends on the type of scanner and device card you use.

You can also use the Scan module to scan hundreds of paper documents for batches and create the BDFs used in Image Pre-Processing, Recognition, and Completion. The BDF contains a list of the image filenames within the batch. You designate how the scanned forms are divided to create BDFs, for example, when scanning a specified number of forms, when encountering a separator sheet, or when identifying a feature such as a bar code.

In addition to creating BDFs, the Scan module can send batches to the JobFlow system, FormWare's workflow module that is used to manage the flow of forms processing. When you add batches to JobFlow, you can assign them a registration or priority number. Using registration numbers, you can group batches together for monitoring or export. Using priority numbers, you can order batches so the most important batches are the first processed regardless of the scanning order. Once the Scan module adds a batch to the JobFlow system, it is immediately routed to the first process in the JobFlow.

The Scan module has the following components:

- **Scan Designer -** The Scan Designer module is a simple, easy-to-use interface that assists you in creating scan configurations. Scan Designer is most typically used by system administrators to configure scanners to work with the FormWare system. The user interface is designed to enable you to access Scan Configuration files (SCG), locate appropriate parameters, modify them, and save the SCG file.

- **Scan Operator -** The Scan Operator module lets you select SCG files, scan paper documents, create BDFs, import image files or faxes, copy existing images from one folder to another, and add batches to a JobFlow.

- **Distributed Scan -** Distributed Scan feeds image and BDF files to a server from remote client workstations. During the distributed scanning process, documents are scanned, and image files and BDFs are created on the remote client workstation. The Remote Transfer

Utility locates the files, transfers them through the Internet FormWare Server (IFS) to the FormWare server for storage, and inserts them into a JobFlow for processing. The IFS and the Remote Transfer Utility control the distributed scanning processes.

ISIS Application Advantages

FormWare Scan seamlessly incorporates Image and Scanner Interface Specification (ISIS®) technology, which facilitates improved communication between a scanner (e.g., a Kodak i840) and FormWare Scan. As an ISIS application, the Scan module supports:

- Video-style and SCSI interface cards
- Maximum-rated scanner speeds
- Recognition of four additional bar codes
  - EAN-8
  - EAN-13
  - ISBN Addon2
  - ISBN Addon5

QuickScan uses Pixel Translations' ISIS libraries to support more than 300 scanners from many manufacturers. ISIS drivers enable scanning at the full rated speed of your scanner. Pixel Translations (a division of Captiva Software) constantly updates the list of scanners supported with ISIS-based applications. For the list of supported scanners, see www.scannerdrivers.com.

Image Processing

Image processing consists of five components:

- Image Pre-Processing
- Image Quality Check
- Form Identification
- Image Enhancement
- Recognition

Each of these components is crucial to the success of your forms processing application.

Image Pre-Processing

Image Pre-Processing (IPP) is a necessary, but often overlooked, part of any forms processing system. It can eliminate many of the problems associated with forms processing applications that use recognition technologies on images. The IPP component cleans up your images to make them "recognition ready." Pre-processing can improve image readability and the recognition accuracy rate of your images.

Deskew and registration are the most important IPP features. The image deskew feature examines lines or text on the image, determines if it is skewed, and straightens the image. When you straighten the image, you can improve the accuracy of the recognition engine. However, to significantly increase recognition accuracy, you should also register the image. Image registration provides consistent upper and left margins for each image. By forcing all images to have uniform upper and left margins, the recognition engine can detect image recognition zones with more precision.

When you have a well-designed paper form, image deskew and registration are the only features you need for precise recognition. However, if your paper forms are not designed for use with recognition technologies, you may need to apply Image Enhancement options. If your paper forms contain lines or boxes that are not in drop-out ink, you may need to remove horizontal and vertical lines during IPP. The IPP component can remove lines that are degraded (contain gaps) and can also reconstruct characters that intersect lines so that the text is maintained for the Recognition process.

IPP can also detect images that are improperly oriented (e.g., upside down) and rotate them to the correct orientation. This prevents unreadable images from proceeding to the Recognition process. You can also rotate images that are scanned at different orientations to avoid an edge perforation. Corrected images overwrite the existing images on disk.

Image Quality Check

The Image Quality Check (IQC) process provides the ability to check and verify the quality of images and identify image problems such as a dark image, light contrast, holes, torn pages, or bent corners. After using either automatic or manual IQC, you can route rejected images to a queue for rescanning or other correction. The two types of IQC are:

- **Automatic IQC** - Runs on images after IPP and before Form ID. It surveys images for problems, and based on defined reject thresholds, rejects those that fail. A BDF with a rejected image (or images) is routed to Manual IQC, where the images are viewed and can be corrected.

- **Manual IQC** - Allows you to manually rescan, rotate, register, or invert images to correct them. You can also perform other corrective procedures, such as reordering the images in the BDF or assigning a data form to an image. Manual IQC is generally run after IPP and Form ID, but can also be run directly after Automatic IQC or after Image Enhancement.

Form Identification

Form Identification (Form ID) is the process of relating production images with a data form using a Form Definition file (FDF). The Form ID component determines which FDF matches an image. The FDF contains the unique recognition attributes for each type of form. If you are using more than one type of paper form, and the paper forms are randomly sorted, the Form ID process allows the forms processing application to instantly recognize each form type. By eliminating the need to manually sort all paper documents before they are scanned, you can decrease your processing costs.

Form ID uses different methods to identify the paper form and load the FDF for key from image or recognition processing. When an image is identified with an FDF, the form number for the FDF is written into the BDF where the image is listed. That form number is used to associate the image with the FDF for processes, such as Completion. There are two basic types of Form ID: Automatic and Manual.

- **Automatic Form ID** - If your application processes multiple types of forms, you can configure Form ID to determine the data form type as the images are processed. You do not need automatic Form ID if you manually batch the same form type together and process only one form type in each run. There are three types of Automatic Form ID:
  - *Dynamic Form ID* uses a set of trained form images to compare to the scanned image.
  - *Feature Form ID* identifies forms based on a specific feature, such as image size, dimensions, or other attribute.

o   *Implicit Form ID* is used with multiple-page documents. The system identifies the first page of the document and then each subsequent page is assigned a pre-defined form number.

- **Manual Form ID** - Batches that fail Automatic Form ID are routed to the Identify Images mode of Completion so an operator can manually identify them. The operator views the unidentified images and types the correct form number. The number is then written to the BDF and the batch can then be routed to Recognition.

Image Enhancement

The Image Enhancement component further enhances pre-processed images to prepare them for the Recognition process. You can run Image Enhancement after you perform IPP on your images or in place of IPP. Unlike IPP processes, which perform functions over an entire image or page, Image Enhancement functions are performed on individual zones on the image, producing an "enhanced image." This allows you to enhance an image at a more detailed level than the page level. The Image Enhancement component offers:

- Image registration and deskew based on an anchored position on the image.
- Line and background removal for individual zones.
- Image enhancement and dot matrix smoothing for individual zones.

When you apply Image Enhancement functions to a zone on your image, the zone is deskewed according to the page level anchors you define for the image, as well as any zone-specific deskew you define for the zone. The Image Enhancement component can also erase image data not contained within a form zone from the processed image.

The basic features within Image Enhancement are Removal and Registration.

- **Removal** - When you define your form, you define zones on the image for the recognition engine to recognize. Selecting the Remove Background option removes all information from the image that you did not define as a recognition zone. This eliminates unnecessary data from the image and reduces the file size of the image.

- **Registration**– You use registration zones to let the recognition engine know where to look for data on each image. The recognition engine checks the X and Y coordinates of these registration zones on each production image and makes adjustments for image variations. Registration consists of three components:
  o   A *registration field* on the data form
  o   A *registration zone* on the image
  o   A *registration anchor* within the registration zone

Recognition

Recognition applies recognition engines to an image to identify characters on that image. Recognition engines convert the machine print text, hand print text, check marks, or bar codes from an image into data in a binary file format, which can be validated using other FormWare processes. Recognition properties are specified on a field-level.

An area on an image, referred to as a recognition zone, is associated with a data field. FormWare Recognition supports the following recognition types: machine print, handprint, mark sense (OMR), bar code, and cursive. For each field you have a choice of one or more of the following recognition engines to execute on your batches: InScript, Nestor, AEG, Mitek, CGK, Parascript, OMR (Mark Sense), and RecoStar Professional.

If you use multiple engines, you can also activate recognition voting, which allows you to apply more than one recognition engine to your data, thereby optimizing recognition accuracy and results. FormWare includes pre-defined voting algorithms for recognition voting. If you want more control over which recognition engine to use, you can also write a special voting DLL in C or C++.

FormWare Recognition provides high acceptance and accuracy rates with rules that let you extract or parse relevant data, modify it, and validate it. Recognition rules use contextual information about the contents of data fields to maximize recognition results. During Recognition, each field is evaluated, processed, and then sent through the rule-based validation, which increases the accuracy of accepted data, minimizes substitution, and determines which fields require reject repair. Recognition rules let you perform *fuzzy lookups* that use wild cards or alternate characters to match or substitute field data. You can write customized rules or requirements for each field.

FormWare also supports Free Form Recognition, which allows you to process variations of a particular form (for example, invoices from different vendors) where the data to be extracted does not always reside in the same location on the forms. Using a keyword search process, you can instruct the recognition engine to "find" the appropriate data.

Full text recognition processes TIFF images of unstructured documents and creates recognition output files in selected formats. It accurately captures text and layout information and exports that text in a variety of electronic document formats. Once information is published to an editable format, it is easy to archive, search on, retrieve, and share throughout an organization via the corporate intranet, the Internet or on removable media.

Completion

The Completion component provides post-recognition processing designed to combine reject repair, image data entry, verification, and data validation.

Reject Repair

Post-recognition reject repair has traditionally been performed in one of two ways: out-of-context repair or in-context repair. Out-of-context repair displays all rejected characters for an entire batch. The operator types only the questionable characters allowing for fast reject repair. However, because the operator cannot view the data in-context, it may be difficult to differentiate between characters such as a capital letter O and the number 0, or the letter S and the number 5. This results in a higher number of incorrect characters inserted into the text.

In-context repair displays the image containing the rejected character in a different color within the recognition zone. This allows the operator to view the surrounding characters and determine the correct value to type in the field. This method is highly accurate but slower than the out-of-context repair.

FormWare combines the speed of out-of-context reject repair with the accuracy of in-context repair. This combination results in a fast, efficient reject repair process. Most of the time, the operator simply examines the out-of-context characters in the reject repair ticker tape and repairs the characters that need correction. However, if the proper correction is not obvious, the operator can look at the in-context view, examine the character and its surrounding text, and type the correct character. This ability to make the right choice the first time eliminates the need for additional processing.

Multiple Passes

The Completion process can be designed to separate the reject repair mode into multiple passes to achieve faster reject repair. This multiple-pass approach to reject repair designates and divides fields by data type, enabling an operator to concentrate on one kind of data.

For example, you could design your Completion process to use four reject repair passes. The first pass repairs only alphabetic fields, the second pass repairs only numeric fields, and the third pass repairs alphanumeric fields. The final pass would be for any remaining non-recognized fields. By dividing the work into separate passes, Completion operators can concentrate on repairing only one type of data and achieve a higher rate of speed and efficiency.

Verification

The Verification mode of Completion offers the ability to verify all fields or only critical fields on your data form. Verification involves an operator retyping the data into the field. As the operator types each character, it is checked against the combination of recognition data and reject repair operator data. If a discrepancy is found, a message displays to the operator. The operator can accept the original data or correct it. Verification is the only way to ensure the complete accuracy of the data and eliminate incorrect character substitutions.

Data Validation

You can also apply validation edits to one or more fields on each data form. Validation edits prevent operators from typing alphabetic characters in numeric fields, lowercase characters in uppercase fields, characters outside of specified ranges, and other data constraints. You enable these edits by turning on a particular validation option. Validation edits are an easy way to run validity checks without the need for custom programming.

JobFlow

The JobFlow system is specialized workflow software for forms processing and contains the following components: JobFlow Designer, JobFlow Client, and JobFlow Monitor. JobFlow automates processing by routing data from task to task without human intervention. These unattended JobFlow tasks perform many of the repetitive chores associated with forms processing. Because forms processing has requirements that are distinct from regular workflow software, the JobFlow system performs the specialized functions necessary for efficient forms processing.

JobFlow Designer

To create a JobFlow system and assign properties to the JobFlow tasks, you use the JobFlow Designer component. Before you set up your JobFlow, it is important to determine the order you want the tasks to run. Batches are assigned to a Job Flow task, or queue, and are routed from one task to another by the JobFlow Client.

JobFlow Client

The JobFlow Client routes images generated by the Scan component through all of the FormWare software processes or tasks in the order they have been set up in JobFlow Designer. For example, you route the images through IPP and Recognition. From there the batch goes to a pre-Completion edit workstation where an edit program is run against the data to reduce the number of substitutions or rejected characters. The batch is then sent to the Completion process where a Completion operator verifies and corrects recognized data. A post-Completion edit workstation

performs a final check to verify any rejected data. Once the Completion process is successful, the data is exported to a database or other back-end repository. Finally, the JobFlow Client runs the cleanup process that removes obsolete data and files from the workstation.

The following sections describe other aspects of JobFlow processing:

Batch Processing

You can set up workstations for processing batches in different ways using the JobFlow system. You can designate a workstation for each task in the JobFlow and then route batches from one workstation to another as the tasks are completed successfully. You can also enable task switching, which will load the different tasks as needed on the same workstation. Besides this program-controlled flow of batches, you can customize the routing for processing errors by specifying a different routing path for batches that contain errors.

Batch Security

The JobFlow system also allows you to control how batches are assigned to Completion operators. When an operator logs into the JobFlow system, certain access rights and permissions are granted and batches to process are assigned based on this information. This automated batch assignment option prevents operators from entering the wrong batch name or accessing the wrong job. The JobFlow Client module assigns the batches, and when the batch is finished, it immediately routes the batch to the next task.

Edit Workstations

The ability to use pre-Completion and post-Completion edit workstations is an important part of the JobFlow system. Edit workstations are attached to the network and poll the JobFlow queue files looking for batches to process. When the edit workstation finds a batch, it runs its edit program against the batch. Edit programs can perform table lookups, data calculations, check-digit calculations, and other mathematical functions on the data within a batch data file. Edit workstations are designed as unattended processes and can run edit programs against batches without human interaction. This allows operators and supervisory personnel to concentrate on more productive tasks.

JobFlow Monitor

If you want to view the status of batches for an entire JobFlow, use the JobFlow Monitor, which displays a visual "snapshot" of the active batches in a JobFlow. All pending, processing, and completed batches display for each specified task. The JobFlow Monitor window allows supervisors to identify processing bottlenecks, reallocate operators or machines to other tasks, and eliminate the roadblock.

Export

Exporting data from FormWare is the final data-capture step. This data, which is maintained in the batch data files, can be exported in a combination of different formats to a text file, a database, a printer, or to other formats.

The following types of export formats are available:

- FPL Export
- ODBC Export
- IMR Alchemy Export

- FileNet Export
- Unisys InfoImage (RBE) Export
- Optika eMedia Export
- XML Export
- OTG AX Export
- VBA Export

System Administrator Responsibilities

To efficiently manage your FormWare system, you need to assign a FormWare Administrator who has the following skills:

- A thorough understanding of the Windows operating system being used.
- Problem solving abilities.
- Good oral and written communications skills.
- The ability to get along with people and be a team player.
- A thorough understanding of the FormWare Administration module.
- A thorough understanding of the FormWare file structure and the files it contains.

System Administration Tasks

The FormWare Administrator must have a thorough understanding of the Administration module and the related JobFlow, JobFlow Maintenance and JobFlow Monitor functions. The administrator must understand the processing cycles, types of files created and used during the processing cycle, and have the ability to navigate the file system to find batches, images, output files, report files, and interrogate system and application error logs. Knowledge of the BDF structure and problem resolution alternatives of exception items is critical to the management of the system. The administrator must understand each of the JobFlow tasks, the Completion and Remote Completion modules, the form types, and the data capture requirements of the application. Captiva offers on-line help, technical documentation and the FormWare File System training class for application designers and administrators.

Various tasks include the following:

- Resetting the Operator Statistics on a regular basis.
- Maintaining the system and user INI files.
- Ensuring the proper data cleanup procedures are performed.

JobFlow Administration Tasks

JobFlow administration tasks include:

- Reset the JobFlow log throughout the day as needed. A minimum of four entries are added to the JobFlow log file for each batch processed by a task. The log file becomes large and unmanageable very quickly and must be renamed or deleted at least once each day. Increased volume may necessitate renaming the log file periodically throughout the day.

- Monitor the processing JobFlow throughout the day. This function is necessary to ensure optimum throughput. JobFlow processes need to be checked periodically throughout the day to respond in a timely manner to resolve any "failed" processes and batch exceptions that may occur. The administrator must be able to access each unattended client process (e.g., IPP, Form-ID, etc.) to start and stop the processes as necessary, and perform any necessary problem resolution

- Use the JobFlow Maintenance Utility to check and/or reset batch status, and route batches as necessary.

- Coordinate the need and assignment of operators for the IQC/Manual Form ID process.

- Reset the JobFlow Monitor counts daily to maintain accurate JobFlow Monitor "processed" counts. It is recommended this activity be done on a daily basis so the JFMonitor function accurately reflects the current processing status throughout the day.

System Troubleshooting Tasks

- Use diagnostic features such as FWDebug to troubleshoot system problems.

- Reproduce any FormWare-related errors and document the steps that cause the problems to occur.

- Use compression software such as WinZip® to zip up the necessary files for reproduction of system problems.

- Communicate with Captiva's Technical Support group for FormWare technical assistance.

Note: Captiva recommends that you train a backup administrator who can perform all of the functions of the System Administrator to assist during emergencies and perform these functions when the System Administrator is unavailable (e.g., off-site training, vacation, sick leave).

Analyzing Form Design

The FormWare software is designed for use with images and recognition technologies. However, just scanning and then performing Recognition on paper forms alone does not result in a significant cost savings necessary to justify the use of recognition technology. To achieve optimum recognition results, you must design forms so that the data can be easily recognized and extracted from the image.

Removing extraneous features, such as lines and boxes, from an image increases the accuracy rate of the recognition engine. Also, when you eliminate unnecessary text from the image, the file size is smaller, thus requiring less disk space and less processing time. The ultimate advantage to a well-designed form, however, is that it improves the recognition accuracy rate.

Electronic forms, referred to as I-Forms, have their own special requirements with regard to good form design. Still essential is data that is easy to extract and export. The variety of form design tools that allow for I-Forms field definition and editing contribute both to the complexity and the ease of using I-Forms.

Analyzing Paper Forms

To determine if a paper form is a good candidate for forms processing using recognition technologies, you must analyze the features of the form. Each paper form has its own characteristics, and these features can impact recognition.

Convincing organizations to redesign paper forms for recognition-based forms processing applications can be a difficult task. Items on the paper form that are aesthetically pleasing to people filling out the form, can reduce the recognition readiness of your forms. For example, if your paper forms contain lines or boxes that are not in dropout ink, you may need to remove horizontal and vertical lines during the Image Pre-Processing step.

As you evaluate a paper form for recognition, consider the following:

- **Paper**
  - Is the paper size supported by the scanner?
  - Is the paper thickness and weight supported by the scanner feeder?
  - Is the color of the paper document either white or a pastel color that can produce a white image?
- **Registration**
  - Does the paper form have marks that you can use for registration?
  - Are items consistently placed on the paper form (such as hand-printed account numbers or imprinted numbers in margins) that can assist in form registration?
  - Does the paper form have at least a half-inch margin on all sides?
- **Form Identification**
  - Does the paper form contain at least one unique feature that is consistently placed?
  - Is this unique feature near the top of the form?
- **Recognition**
  - Do hand print fields on the paper form contain segment lines, combs, or boxes?
  - Are boxes for constrained hand print fields in dropout ink?
  - Are the lines for the boxes, if not in dropout ink, at least 1/16 of an inch thick?
  - Do the hand print boxes have an interior size at least .30 x .30 inches (1/3" x 1/3")?
  - Is the vertical spacing of the constrained hand print rows at least the same as the height of the boxes?
  - Do the mark sense boxes have an interior size of .25 x .25 inches (1/4" x 1/4")?
  - Is the vertical and horizontal spacing of the mark sense boxes at least the same distance as the width and height of the boxes?
  - Are the field labels on the image at least .25 inches (1/4") away from the text to be recognized?
  - Are all hand print boxes the same size?
  - Are the machine print characters in a sans serif font with uppercase alpha characters?
  - If characters are numeric and machine printed, are they printed in a trained font such as OCRA, OCRB, MICR, or Farrington?
  - Are unconstrained, hand print areas boxed and at least one inch away from any data to be recognized?
  - Is the point size of machine print characters 10 or 12 point?

      o   For free-form fields, are there suitable keywords upon which the recognition can search?

      o   Are intra-field separators, such as periods or commas, in dropout ink?

If you answered "no" to most of these questions, your paper form is probably not ready for processing using recognition technologies. Instead, it would be a candidate for manual data entry (key-from-image) until you can redesign the form contents and structure.

If you answered "yes" to most of these questions, your form is a good candidate for forms processing with recognition, although additional Image Pre-Processing steps may be required before performing recognition. An important step in the form design process is, after creating the form, testing the application to determine if the paper form truly is recognition-ready.

Elements of Good Paper Form Design

The following guidelines for designing paper forms are true for most paper forms that you process using recognition technologies.

- Use white paper. A shaded or colored paper can increase the speckles or noise on the form and reduce recognition.
- Place a horizontal line at the bottom of the paper form. This clearly delineates the end of the form and improves image registration.
- Put understandable processing instructions on the front top of the paper form. Instruct the person filling out the form to type the data, follow the guidelines, and use white correction fluid for mistakes. Also, put a statement on the form that directs the person to fill out and return the original form, not a photocopy.
- Use carbon-less dropout ink for instructions and captions. This reduces clutter on the form image and the amount of disk space needed to store the image.
- Use dropout ink for the boxes into which data is typed.
- Provide instructions that tell the person filling out the form to put hand-printed information in uppercase letters only. Allowing both uppercase and lowercase letters on the form decreases the recognition accuracy rate.
- List each field name and instructions for completing the field. For example, on a date field, use the caption: "Enter Date (DD/MM/YYYY)." On a name field, use the caption: "Enter the patient's first name, middle initial, and last name."
- Make all boxes in which data is entered on the paper form at least one-half inch from the edge of the paper.
- Place signature lines or other unconstrained handprint areas as far as possible from all other data fields. This prevents users from inadvertently writing data into nearby fields.
- On the back of the paper form, in dropout ink, place more detailed instructions for completing the form. You can also include a phone number that the person filling out the paper form can call when questions arise. Be sure to refer to these more detailed instructions on the front of the paper form.

Analyzing I-Forms

Designing good electronic forms has certain of the same elements as paper forms, such as providing instructions for completing the form. However, references to pages, dropout ink, typing, photocopying, etc. are not relevant for I-Forms.

Follow these guidelines as you create your I-Form:

- If you are building a form from scratch, spend time planning the data fields the form should contain.
- If you are building a form from a paper model, evaluate what and how you should modify it to take advantage of electronic features and functionality.
- In planning the form layout, consider space issues, such as maximum field length for each field, sequence of fields on the form, and groupings, if applicable.
- Design field labels to clearly indicate the type of information you want to capture in each field.
- Determine which fields are required and which are not required.
- Consider how to alert users of the required fields on the I-Form. For example, when the user leaves the required field or the form, display a message that reads: "This is (or Name of the field) a required field. Please enter the requested information." When the user clicks the OK button, the cursor returns to the field that requires data.
- Determine the field types appropriate for the types of data on the form, e.g., text or number fields, combo boxes (drop-down menus), check boxes, or radio buttons.
- Determine how to notify users if they type invalid values into a field. For example, when the user attempts to leave the field, display a message that reads: "You have entered invalid characters in this field. Please correct." When the user clicks the OK button, the cursor returns to the field.
- Test the draft version of your I-Form by selecting the View in Browser button.
- Fine-tune the I-Form and usability-test it before final publication.

Note: Remember that the display of the form is Browser-dependent. Some design features and formatting will be significantly different when rendered in one browser vs. another. Therefore, if you can test forms using different browsers, you should be able to create a valid end product.

Form Design Resources

Many options are available to assist you with your forms analysis and design efforts. Captiva Software Corporation offers forms design consulting services. Captiva personnel have many years of experience in forms processing and designing paper forms that make the best use of recognition technology. A consultant can work with you to review your current forms and then propose changing these forms—or redesigning them as I-Forms. This redesign process can dramatically improve throughput and recognition accuracy. For additional information on consulting services, contact the Captiva sales office at 858-320-1003.

Several books and other resources are available on the elements of form analysis and design from the Association for Information and Image Management (AIIM) on their web site, *www.aiim.org*.

Planning Your Project

Before You Begin

Before you begin implementing FormWare File System, you need to analyze how it will work in your environment and how you will configure it to meet your unique business requirements. Each project plan should include functional requirements of the planned application, detailed specifications of that application, and an acceptance test plan.

Before you start designing with FormWare File System, you need to have a complete understanding of the requirements and answer the following questions:

- Have I obtained the proper FormWare File System training?
- Have I read the FormWare File System documentation completely?
- Have I set realistic time frames for implementation?
- Will I need Captiva's Professional Services team to assist me?
- Where will my documents come from?
- What types of documents will I be processing?
- Will I be processing multi-page documents?
- Will I be processing attachments with those documents?
- Do I need to capture any information from attachments?
- How many documents will I process per day?
- Will my mailroom need to sort or add separator sheets?
- What kind of scanner should I buy?
- What card, cable, and drivers do I need for that scanner?
- Does my scanner have a dropout bulb?
- Does my scanner imprint or annotate?
- Will I need to imprint/annotate/store a Document Control Number with every image?
- What is the quality of the images I receive/scan?
- Do I need to adjust my images during Image Pre-Processing to make them uniform?
- Do any of the documents I receive have stamps or writing that might affect processing?
- Where will I export the information I capture?
- What format does my export data need to be in?
- On which forms will I perform Recognition?
- Do I have any requirements for Recognition accuracy?
- Do I have any requirements for field-level and document-level accuracy?
- On which forms will I perform Key From Image?
- Do I need to capture data for every field on the form?
- Do I understand the business rule that should be applied to every field I capture?
- Will my business rules use a lookup table?
- Do I have access to data to populate lookup tables for my application?

- Will I need to write custom business rules and edits?
- Will I need additional training/professional services to write business rules and edits?
- Will I need a data entry operator to perform additional verification of any fields?
- What tasks do I need in my JobFlow?
- Where will I install FormWare File System during development/testing?
- How will I obtain a test suite of images?
- Who will administer the application after the implementation?
- How will I train the FormWare administrator?
- How will I train the data entry operators?
- How will I train the scan operators?

Functional Requirements & Detail Specifications

Before implementing FormWare File System, you should write a document that provides a high level description of the proposed system and information necessary to implement that system. It should include a description of the functional requirements, detailed specifications, hardware components, applicable software, and reporting capabilities. It should also identify the parties responsible for the various components of the total solution.

Your planning document should provide the following information:

- An overview of what the software can do.
- The document types that will be processed on the system.
- The hardware configuration that will accommodate the expected peak daily volumes and your business rules.
- The operation and limitations of subsystems.
- Detailed descriptions of application processes.
- Data capture and edit procedures that will achieve your processing requirements.

This specification should include a description of the applicable software; job, form and field properties; JobFlows; operator procedures and statistical reports; output programs and procedures; and administrative requirements.

Following is a sample list of items that you need to define in your document.

1. **Introduction**

   a. **Purpose** – Provide a description of the reason for the document, to solidify the functional design of the proposed application. Also list information the document includes.

   b. **Scope** – Provide a high-level description of the proposed system. Describe the hardware components, applicable software, and reporting capabilities. Identify the parties responsible for the various components of the total solution.

   c. **Applicable Documentation** – List documents relevant to FormWare File System. For example, user documentation, internal documents, and data documents.

   d. **General Constraints** – List any prerequisites, general requirements or restrictions to which the proposed system must adhere.

   e. **Testing/Development Media Requirements** – Describe the acceptance test plan and the images needed to perform the testing.

    f. **Source Media Descriptions** – List the document types and volumes the application will process. Additionally, describe the source of the documents, how to prepare documents for scanning, and batch preparation requirements.

2. **JobFlow Description and Requirements**

    a. **JobFlow Process** – Describe the flow of work through the system. If the JobFlow is to be controlled by a product other than FormWare JobFlow, state it here.

    b. **Process Flow Chart** – Create a diagram of the JobFlow process.

    c. **JobFlow Properties** – These properties control the automatic routing of batches. Define the properties of each task here.

    d. **Task Switching** – Define the JobFlow tasks that will run on each client machine, including the intervals for the tasks.

    e. **Execution Procedures** – Describe how system administrators will begin batch processing.

    f. **Exception Conditions** – Describe how exceptions (e.g., physically damaged documents, edit failure processing, rescan processing) will be handled.

3. **Detailed Job Specifications**

    a. **System Defaults** – List the system default properties to be used with the application.

    b. **Job** – Describe the forms, job properties, database requirements, and form properties of each form used in the job.

4. **Administrative Specifications –** Describe the files included and where they are stored in the \\FWShare\FormWare directories.

    a. **Files** – Contains subdirectories with production batches and batch definition files.

    b. **Forms** – Contains the productions forms created in Designer. The forms have an FDF extension, and the help files have an HLP extension. The setup and overlay images are stored here too.

    c. **Images** – Contains production images and subdirectories with production images.

    d. **INI** – Contains subdirectories with INI files.

    e. **Index** – Contains index files created for table lookups. Index files require a corresponding data file that is listed in the Files Folder section.

    f. **Jobs** – Contains the jobs created. Jobs have .JDF extensions, and the help files have .HLP extensions.

    g. **Lists** – Contains the dictionary files and the predefined list of field values.

    h. **VBA** – Contains the Visual Basic project files.

5. **System Architecture**

    a. **System Diagram** – Provide a topography of your system.

    b. **Hardware Components** – Provide a brief functional description of each piece of equipment necessary for proper system operation. The more complex the requirements, the more horsepower the workstations should have. List who is responsible for each component's operation.

       o Scanners

       o FormWare File Server

- o   Internet FormWare Server
- o   Image Pre-Processing Stations
- o   Recognition Stations
- o   Edit Stations
- o   Completion Clients
- o   Export Stations
- o   Network
- o   Tape Drives
- o   UPS (Uninterruptible Power Supply)

c.   **Software Description** – List the proposed software and provide a brief functional description, including the operating systems to be used on the servers and workstations.

d.   Network Communications

- o   **Communications Protocols** - Describe the communication protocols to be used. State who is responsible for network reliability and performance.
- o   **Mainframe Connectivity** - Describe the requirements for connecting to the host as it relates to the workflow requirements.
- o   **External Connectivity** - Describe the requirements for connecting to any other system as it relates to workflow.

e.   **Security** – Briefly describe the security requirements for system access, including security levels (e.g., operator, supervisor, system administrator), password protection, operator IDs, etc.

f.   **Disaster Recovery** – Describe what occurs if there is a power failure, including how much data is lost. Explain how data is recovered. Outline possible batch states and identify applicable steps required to recover affected batches/records. List each software component with appropriate recovery procedures if necessary.

Acceptance Test Plan

The acceptance test plan is used to verify that all aspects of the system have been implemented as designed. You must design all test scenarios to map back to the requirements agreed to in the written functional requirements and detailed specifications.

All acceptance tests should simulate the anticipated production environment as closely as possible. The tests must be executed on the actual production hardware. All software applications that will be used in production should be loaded on the system for the most effective and successful acceptance test results. In addition, it is essential to simulate a production volume stress test for all types of form processing that will occur in production. These precautionary testing procedures will detect unexpected hardware, software, or volume related problems that do not become apparent during the development process.

Refer to the Developing an Acceptance Test Plan section for more information.

2 *Installing & Configuring*

Installing & Configuring Overview

This section contains instructions for installing FormWare File System software and configuring any settings necessary to prepare for designing an application. The following topics are addressed in this section:

- Understanding System Architecture
- Configuration Guidelines
- Setting Up Hardware
- Setting Up a Scanner
- Installing the software, which includes:
  - o Installing the File Server
  - o Installing Client Workstations
  - o Installing the Security License Manager
  - o Installing QA Statistics
  - o Installing Scan Drivers
  - o Installing Adobe Acrobat Reader
  - o Installing Components for Distributed Processing
  - o Installing the Support Utilities
  - o Modifying Your Installation
- Establishing Security
- Configuring the software, which includes:
  - o Configuring the File Server for Distributed Processing
  - o Configuring the IFS
  - o Using Remote Scripting
  - o Connecting a Remote Client to a File Server
  - o Configuring the Software
  - o Setting Client Display Properties
  - o Using Desktop Shortcuts to Launch File Server Components

Understanding System Architecture

The FormWare File System software is a collection of programs designed specifically for the Windows environment. In addition to its graphical user interface, the software is modular, allowing you to purchase only the specific modules you need. By understanding the modular architecture of FormWare, you can design and operate a forms processing application tailored specifically for your organization.

The FormWare software is designed to process high volumes of forms and typically uses a network to achieve this goal, which maximizes efficiency by spreading the work across multiple machines that can process the work simultaneously. At the heart of the system is a FormWare File Server that, along with the network, must be capable of handling peak loads of images and data without significant processing reduction. Attached to the network are workstations that perform the forms processing tasks.

Network Configuration

The following diagram shows a typical network configuration for a FormWare File System.
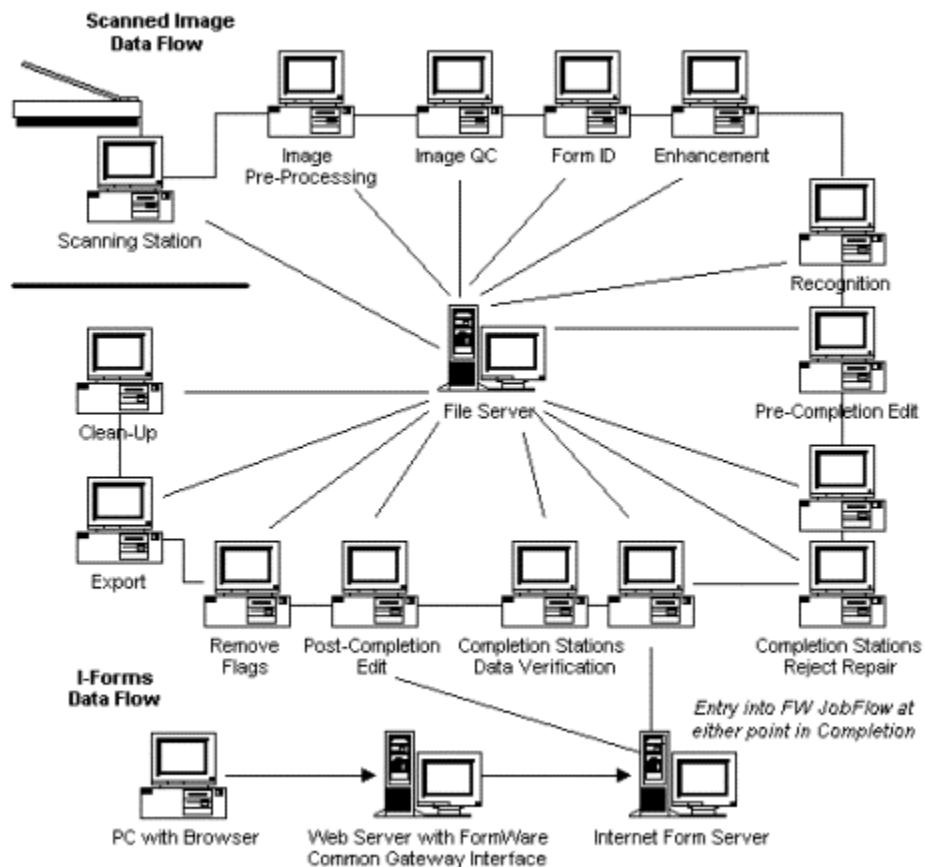
The diagram indicates one or more workstations for each module. However, the number of workstations necessary for Scanning, Image Pre-Processing, Recognition, Completion and other tasks depends on the number of forms you process each day. Your system can have more or fewer workstations than illustrated, depending upon the amount of processing and editing required for your forms processing application. Using a network and the FormWare software, you can create an efficient forms processing solution.

The FormWare approach to forms processing is to divide and conquer. Because processing is divided among several workstations dedicated to separate JobFlow tasks, if a bottleneck occurs for a particular task, such as Recognition, you can switch the Image Pre-Processing station to perform Recognition and eliminate the processing traffic jam.

Domain Controller

Captiva recommends that you set up a single domain and implement two domain controllers, one for production, the second for production backup. Active Directory resides on the domain controllers in a native Windows 2000 domain. In a Windows 2000 domain, the File Server does not have to be on the domain controller.

Types of Environments

FormWare can be set up to operate in separate environments, as follows:

- Development
- Test
- Production

A new customer who has purchased the FormWare File System can follow the installation instructions provided in this section. Once the application is installed, you can create the development, test, and production environments by setting up different JobFlows within the Designer module. FormWare typically works with complementary software packages and hardware components.

Types of Servers

A customer using FormWare distributed technologies (see the Installing Components for Distributed Processing section for more information) and processing any substantial volume of work through its Web Server should setup three separate servers as listed here:

- Internet FormWare Server (IFS)
- Web Server (running IIS and FormWare Common Gateway Interface)
- File Server

The above three-server environment will enable workload sharing in the event of potentially heavy workload across multiple machines rather than risking a bottleneck by consolidating all of these tasks to a single computer.

FormWare File Server

Install the FormWare software on a separate file server machine. On the server side a domain is recommended. On the remote client side, no domain is required. The FormWare File Server need not be on the domain controller in a Windows 2000 domain. Although it is possible to install the FormWare File Server on a domain controller, it is best in a production environment to separate these two machines for load balancing.

IFS

The Internet FormWare Server (IFS) prepares data for submission into a JobFlow and is the liaison between the web server and the File Server. All communication between remote clients and the File Server is routed via the HTTP protocol through the Web Server to the IFS. The IFS is only necessary on the host site if you are installing FormWare Distributed Processing.

Web Server

The Web Server is only a necessary component on the host site if you are installing FormWare Distributed Processing. It accepts HTTP requests and processes any FormWare-specific requests via FWCGI (on the WWW server). FWCGI communicates the request to the IFS, which is the interface to the FormWare File Server. IFS builds a valid HTML response, which is passed to FWCGI for transmission.

Client Workstations

Set up the client for a user with:

- Administrator rights to install FormWare
- A valid domain account (rights to use in set up for clients that will be running unattended; i.e., the computer can process applications with no one logged on)
- Anti-virus software either disabled or set to run in auto-protect mode (Norton)

Configuration Guidelines

The FormWare System (File or Database) may be set-up using a variety of PCs and LAN hardware. While there are very few specific system requirements, depending upon your application (images per day, number of characters, FormWare modules installed, etc.), in order to optimize overall performance, there are various processing and storage elements that need to be examined.

In an effort to assist you, we have prepared some basic guidelines for configuring/optimizing your FormWare data capture system.

Caution: Running anti-virus software on FormWare servers and client machines will affect the performance of the Database application. Certain settings in your anti-virus software may make FormWare unusable. For example, if you are running Norton Anti-Virus and have the program set to Auto-Protect, Norton will check every file on the workstation during processing. Norton recommends that you run in Auto-Protect mode unless you have been directed to disable it by a software package that you are installing.

Enterprise Requirements and Recommendations

This section lists the system requirements and recommendations for the computer running the Enterprise Server

Enterprise Server Software Requirements

The Enterprise Server system is designed to operate with Microsoft Windows. You may run the FormWare application under the File System, Database or InputAccel with the following:

FormWare File Server

- Microsoft Windows 2000 Server/Advanced Server with Service Pack 3 or later.
- Microsoft Windows NT 4.0 Server with Service Pack 6a or later.

FormWare Database Server

Enterprise

- Microsoft Windows 2000 Server/Advanced Server with Service Pack 3 or later.
- Microsoft Windows NT 4.0 Server with Service Pack 6a or later.

AND

- Microsoft SQL Server 7.0 with Service Pack 3 or later.
- Microsoft SQL Server 2000 with Service Pack 3 or later.

InputAccel Server

High Volume Installations (>10,000 PPD)

- Microsoft Windows 2000 Server/Advanced Server with Service Pack 3 or later.
- Microsoft Windows NT 4.0 Server with Service Pack 6a or later.

Low Volume Installations (<10,000 PPD):

- Microsoft Windows 2000 Professional with Service Pack 3 or later.
- Microsoft Windows NT 4.0 Workstation with Service Pack 6a or later.

Note: Captiva recommends using the Server versions of the Windows products whenever possible to provide maximum performance.

Enterprise Server Recommendations

For best performance, we highly recommend a dedicated single Windows NT or 2000 server to run the Enterprise Server. Due to the way Windows switches tasks, running additional modules on the same machine as the Enterprise Server can result in a significant performance drop, even if the additional modules do not use much CPU time. Additional recommendations are listed in the sections that follow.

Disk System

The speed of the disk I/O subsystem used by FormWare often proves to be the most important performance characteristic on high volume FormWare installations. There are many high-end disk controllers and disk systems that should provide excellent performance. Rather than focus our testing and recommendations on specific disk controller options, we instead offer some tips in purchasing and configuring the disk system for the best performance for FormWare and the Enterprise Server system.

We recommend that you use trusted and reliable suppliers of disk system components. For high volume systems you should buy the equipment that promises to deliver the highest performance. The drives should be high RPM, high capacity, SCSI units.

The FormWare File Server should include hardware level support with 10+ GB fault tolerant RAID Level 5 disk controller(s) using high performance SCSI hard disk drives. The FormWare Database Server should included hardware level support with 20 300+ GB fault tolerant RAID Level 5 disk controllers using high performance SCSI hard disk drives. The InputAccel Server should include hardware level support with RAID 0+1 disk controllers.

Database System

Depending on daily volumes and number of clients, dual or quad processors will be required. In large configurations the Statistics database and QA Stats Database may be moved to a separate server for optimum performance. In that case, two SQL servers would be required, one for the FormWare WorkFlow Database and the other for the Statistics Database.

There are two versions of Microsoft SQL Server that Captiva Software supports, Microsoft SQL Server 7.0 (Service Pack 3 or later) and SQL Server 2000 (Service Pack 3 or later). However, we recommend that the client purchase MS SQL Server 2000 for new installs. Captiva Software does not resell SQL Server. Both SQL Server and Client licenses need to be purchased through Microsoft or one of their many resellers. FormWare Database also requires Internet Explorer 5.01 or later for the server and subsequent clients.

The amount of hard disk drive space required for the database server will vary greatly, depending upon the size and volume of data processed on a daily basis and the number of days the data is retained. Note that the actual TIF or JPG images are NOT stored in the SQL Server, but are stored in the FormWare file server. The Workflow Database contains the data item hierarchy and the OCR and batch data. The customer's database administrator should be consulted to properly size the server but the following are some guidelines you can follow for the WorkFlow Database, Statistics Database and the QA Statistics Database

Determining SQL license requirements

Captiva Software does not resell SQL Server. Both SQL Server and Client licenses need to be purchased through Microsoft or one of their many resellers. For basic pricing information for SQL Server and Client, refer to the following web site:
http://www.microsoft.com/catalog/display.asp?site=10145&subid=22&pg=8

InputAccel System

FormWare can work under InputAccel. Again, the drives should be high RPM, high capacity, server class SCSI units. The disk controller(s) should include hardware level support for RAID 0+1,provide on-board disk caching of at least 16MB, allow for Write-Back (write to RAM) caching, and provide battery-backup for the on board cache.

When installed, we have found that RAID 0 (data striping) offers considerably higher performance than RAID 5 (redundant parity drive) systems. If data redundancy is a concern, we recommend using RAID 0 in combination with RAID 1 (mirrored data). Often this is referred to as RAID 0+1. (Please refer to the RAID Configuration Performance addendum to this document for more information). We do not recommend using the native NTFS mirror sets, strip-set or stripe-set with parity configurations available through Windows Disk Administrator.

The disk controller card should be configured to use Write-Back caching rather than Write-Thru caching. Write-Thru caching slows disk I/O by waiting until data has been physically written to disk before completing a write operation. With Write-Back caching, the I/O operation completes almost immediately as the controller card caches the write operation in it's own RAM so that it can be written to disk at a more convenient time.

Internet FormWare Server

Captiva's Internet FormWare Server (IFS) Module empowers businesses with another data capture channel — the Internet — to extend their information gathering to end-users and customers. Now, forms filled in on the Internet, and XML data streams, generate data records that can be submitted to FormWare's WorkFlow for processing with data captured from paper, images and fax. Companies can also use the Internet FormWare Server to tap the Internet's cost-

effectiveness as a data ramp for distributed processing such as remote scanning, remote data correction/entry operators and browser based data entry. The IFS provides several features which include:

- Internet Forms data capture & XML data streams
- Distributed Scanning
- Distributed Completion
- Distributed WorkFlow
- eEntry (Browser Based Data Entry)

FormWare's Remote Processing components combine certain client processes with the functionality of HTTP client services. Using the Hypertext Transfer Protocol (HTTP), provides added performance benefit. It is an application-level protocol with the speed necessary for distributed systems and is based on a request/response paradigm. The HTTP protocol is connectionless and stateless. After the server responds to the client request, the connection is not only dropped but also forgotten. Each request is treated as a brand new request without any context. This allows a larger pool of people to work from a smaller group of physical connections and uses bandwidth more efficiently. However, it also demands that the IFS provide session management for multi-page I-forms document creation.

FormWare's Internet solution supports a Public Key Infrastructure when deployed using SSL technology on either Microsoft's IIS web server running on Windows NT or 2000, or Apache (version 2.0) Linux as the web server.

Internet FormWare Server supports any browser that supports HTML 4.0 (DHTML) and JavaScript. This includes Internet Explorer 6.x and later (for Windows platforms) and Netscape 4.x and later. However, be aware that eEntry uses component technology to provide the image viewport in the browser. Thus, the browser being used must support ActiveX controls for proper eEntry functionality.

RAM

The more RAM the better, but 1 GB is a good starting place. As the load on the system increases, you should use the Windows Performance Monitor (PERFMON) utility to track how often the server swaps to disk and to track free memory resources while the system is running.

We suggest that you purchase the fastest and most robust system available in the market place today when possible. The minimum recommended server requirements are as follows:

FormWare Server	Pentium Processor Speed	RAM	Operating Systems
File System	800 MHz, 1.8 GHz recommended	128MB minimum, 256+ recommended	Windows NT 4.0 (SP 6a), and Windows 2000 (SP3)
Database	1.2 GHz, 2GHz recommended	768MB 1.5 GB RAM, with integrated cache	Windows NT 4.0 (SP 6a), and Windows 2000 (SP3)
InputAccel	1.2 GHz, 2 GHz recommended	768MB	Windows NT 4.0 (SP 6a), and Windows 2000 (SP3)
Panagon	1.2 GHz, 2 GHz recommended	768MB	Windows NT 4.0 (SP 6a), and Windows 2000 (SP3)
Internet FormWare Server (IFS)	600 MHz minimum, 800MHz recommended	128MB minimum, 256+ recommended	Windows NT 4.0 (SP 6a), and Windows 2000 (SP3)

FormWare Server	Pentium Processor Speed	RAM	Operating Systems
Web (IIS or Apache)	600 MHz minimum, 800MHz recommended	128MB minimum, 256+ recommended	Windows NT 4.0 (SP 6a), and Windows 2000 (SP3)

Desktop Requirements and Recommendations

The FormWare Desktop Edition is supported on three (3) platforms: FormWare File System, FormWare Database System and FormWare under InputAccel. We recommend that you purchase a system designed for optimization.

The FormWare Desktop also allows users to add-on a limited number of additional modules. The licensing mechanism in the FormWare Desktop will search for the first license which resides on the desktop machine and then will look for any additional add-on modules.

Note: If add-on modules are being used, then the last component of a given type in the desktop configuration to be started must be started on the desktop machine itself. For example if you have 1 completion license on the desktop then 2 additional completion licenses, the 2 add-on completion license should log-on at the beginning before the completion license on the desktop. However, if one of the non-desktop Completion licenses log out, it will not be able to log in again until the desktop user also logs out – because the last license used must be allocated to the desktop.

FormWare File System

- Microsoft Windows NT 4.0 Workstation with Service Pack 6a or later (one per client)
- Microsoft Windows 2000 Professional with Service Pack 3 or later (one per client)
- Microsoft Windows XP Professional with Service Pack 1 or later (one per client)

FormWare Database

- Microsoft Windows NT 4.0 Workstation with Service Pack 6a or later (one per client)
- Microsoft Windows 2000 Professional with Service Pack 3 or later (one per client)
- Microsoft Windows XP Professional with Service Pack 1 or later (one per client)

AND

- Microsoft SQL Server 7.0 with Service Pack 3 or later.
- Microsoft SQL Server 2000 with Service Pack 3 or later.

InputAccel System

Low Volume Installations (<10,000 PPD):

- Microsoft Windows NT 4.0 Workstation with Service Pack 6a or later (one per client)
- Microsoft Windows 2000 Professional with Service Pack 3 or later (one per client)
- Microsoft Windows XP Professional with Service Pack 1 or later (one per client)

Module Requirements and Recommendations

This section lists the system requirements and recommendations for the client computers running FormWare modules.

FormWare Module Software Requirements

- Microsoft Windows NT 4.0 Workstation with Service Pack 6a or later (one per client)
- Microsoft Windows 2000 Professional with Service Pack 3 or later (one per client)
- Microsoft Windows XP Professional with Service Pack 1 or later (one per client)

FormWare Module Recommendations

The recommendations provided below can be used to assist you in specifying new systems for purchase. It is always highly recommended that you purchase the fastest and most robust system available in the market to day when possible. If you will be using existing computers as FormWare client workstations, each must meet or surpass the minimum supported configuration.

Tasks that FormWare modules perform on are categorized into CPU Intensive and Less CPU Intensive groups to help you properly match computer power to module tasks. These tasks are listed below.

CPU Intensive Modules

- Scanner Control (Scan and Rescan)
- Image Pre-Processing
- Dynamic Form Identification
- Image Enhancement
- Color Image Pre-Processing
- Recognition
- Virtual Classification
- Automatic IQC
- Full Text PDF

Pentium Processor with 2GHz or faster recommended (800 MHz minimum), 256-512MB RAM, 4-8 GB Hard Disk Drive, Internet Explorer 5.01 or later (Required for Database installations only), Minimum of 250MB of free space, and CD-ROM Drive.

Less CPU Intensive Modules

- Designer
- Administration
- Completion
- Manual IQC
- VBA Edit
- Export

Pentium Processor with 1GHz+ recommended (600 MHz minimum), 128 MB for File System, 256MB+ for database RAM, Internet Explorer 5.01 or later (Required for Database installations only), Minimum of 250MB of free space and CD-ROM Drive.

Additional Scan Module Requirements

The FormWare Scan Module has been re-engineered to use Pixel Translations PixTools and ISIS scanner drivers. As a result, FormWare Scan 5.0 supports more scanners and more interfaces—including USB, 1394 and Adaptec SCSI.

Earlier versions of the FormWare Scan Module were based on the Kofax ImageControl toolkit (Software), which required users to purchase a Kofax card (Hardware) or software to communicate with their scanners. With the new version of the FormWare Scan Module, a Kofax card is no longer required for SCSI scanners. For users who wish to take advantage of their existing Kofax Adrenaline Video cards, FormWare Scan 5.0 supports ISIS drivers that communicate with the Kofax Adrenaline 850V and 1700V cards. The Adrenaline 850S and 1700S are no longer supported. FormWare Scan 5.0 with ISIS comes with four (4) scanner levels that are controlled by the License Manager:

- Level 1 – Personal
- Level 2 – Workgroup
- Level 3 – Departmental
- Level P – Production (e.g., Kodak i800 series)

Note: For information about the scanners and SCSI Adapters that Captiva Supports, see the PixTool's web site at http://www.pixtran.com.

Scan Client (Kofax Version)

For existing customers who currently have the FormWare Scan module based on Kofax's ImageControls, Captiva still offers this module, but with functionality frozen at the feature set of the FormWare 4.6 Scan module. This module will not be enhanced in the future to support newer Kofax software. It requires a Kofax supported scanner and one following combinations of hardware and third-party software products:

- A Kofax Adrenaline interface board, Adrenaline cable and Adrenaline software driver, and a supported scanner or
- An Adaptec 1542 or 2940 SCSI interface board, Adaptec SCSI cable and a Kofax Adrenaline Capture Engine (ACE – without bar code & patch code support – approximately $300) or a Kofax Adrenaline Image Processing Engine (AIPE – with bar code & patch code support – approximately $800), both of which include ISIS software drivers and are required regardless of what software drivers you may have received from the scanner manufacturer and a supported scanner.
- Kofax Image Controls version 3.3 or later - version 3.6 or later is recommended.

HTML Help System

In order for end users to view and run the FormWare Microsoft HTML Help system, you must have Internet Explorer 5.0 or later on each system running a user interface task. This Web browser is built into the Windows NT, Windows 2000, and Windows XP operating systems. However, using later versions of IE (such as version 6.0) will enhance the user's HTML help experience by providing the most current support for the HTML Help components.

Network Recommendations

It is recommended that a FormWare system be set up as a discrete sub-network that is isolated from other LAN applications/traffic. This is especially true for high volume applications where network utilization can be high. Captiva also recommends the use of 1 to 4 100Mbps (or faster) network adapters (Ethernet network interface cards) with 100 Base-T switches are required in order to provide greater bandwidth and optimal throughput. All file servers must be plugged into an uninterruptible power supply (UPS) that provides both uninterruptible power and protection against power surges. All hubs and processing workstations should also be UPS protected.

We recommend no more than 50 PCs per isolated network segment. However, scanners, output devices, and unattended processes requiring image files (like image pre-processing and recognition) generate substantial network traffic and require special consideration. We recommend all these devices or processes be installed or run on PCs attached to the same isolated network segment in order to isolate the related network traffic and avoid system-wide performance degradation. PCs for data entry operators may be attached to the same network segment, but you may not be able to attach a total of 50 PCs without experiencing an unacceptable level of performance degradation. In addition, the tasks being performed by data entry operators, the productivity of those operators (both of which in turn determine how often image files are accessed), and the size of the related image files, may limit the number of data entry operator PCs that can be attached to the same network segment.

Network bandwidth requirements for the FormWare workflow database can be double the requirements for the FormWare File System due to SQL query overhead. It can be more when WorkFlow or QA statistics are enabled. Please plan accordingly when estimating traffic requirements.

Installing the Hardware Keys

Understanding Hardware Keys

Hardware keys are dongles that interface with the Parallel/USB ports on the computer and can be programmed by Captiva to allow you access to FormWare features. They determine which components you can run and the performance levels allowed, such as characters per day (CPD) on a recognition engine. The FormWare Security License Manager (SLM) governs what components you can use in your system. The hardware key drivers accommodate both parallel-port and USB keys. Your packet should contain keys appropriate to the software you have purchased and the environment into which you will install that software. Depending upon the features you purchased and to which you will need access, you will have a subset of the following hardware keys:

- Blue HASP parallel-port key - Attaches to the parallel port on the computer where FormWare SLM is running.
- Black HASP USB key - Attaches to the USB port on the computer where FormWare SLM is running.
- Purple USB Rainbow Superpro key - Attaches to the USB port on the computer where FormWare SLM is running.
- Gray parallel-port key - Attaches to the parallel port on the computer where FormWare SLM is running..

Note: The FormWare software supports USB keys only on Windows 2000 and Windows XP machines. If your host site is based on Windows NT4, your security hardware key must be on the parallel port, because NT4 does not support USB. Do not try to attach a USB key to an NT4 machine.

Important Notice: For customers enrolled in a Captiva AdvantEDGE maintenance program, any damaged or malfunctioning key returned to Captiva will be replaced free of charge. Lost or destroyed keys will not be replaced free of charge and must be repurchased for the suggested list price of the related software.

Installing Keys for a New Installation

Before installing FormWare on your computer for a new installation, attach the parallel port keys to the parallel port on the back panel of the computer chassis. If you use USB keys for licensing, remove them and re-attach them only when instructed to do so during the installation procedure that follows.

Converting to HASP USB for Existing Installations

If the software is already installed and you are converting to USB-controlled SLM, do the following:

1. Stop all processes on all machines in the Database system.

2. Go to the Windows Services program on the FormWare File Server or client.

3. Right-click the Security License Manager and select **Stop**.

4. Remove the parallel port HASP key(s).

5. Attach the USB key. The Windows hardware wizard launches and recognizes the USB key as a new device. The wizard auto-installs the driver for the new USB device.

6. Return to Windows Services and restart the Security License Manager.

Setting Up a Scanner

Understanding Scanner Types (ISIS)

Because ISIS is a de facto standard and the basis of an industry standard (ANSI/AIIM MS61-1996), ISIS users are assured that their imaging applications will drive a wide variety of scanners. The list of supported scanners is lengthy and changes continually. Obtain an up-to-date list of supported ISIS scanner drivers at www.scannerdrivers.com, a Captiva web site that is organized as follows:

- **Driver List** - Complete list of scanners in the Captiva database, listed by type, driver name, level, and version platform
- **Scanner List** - Complete list of drivers available for purchase
- **Latest Drivers** - List of recently released, ISIS-certified scanner drivers
- **ISIS** - Definition and overview of ISIS

Understanding Scanner Types (Kofax)

The following tables offer a compatibility matrix of scanners and Kofax Adrenaline scanners and connectivity.

Scanner	Kofax Adrenaline interface board, Adrenaline cable and Adrenaline software driver	Adaptec 1542 or 2940 SCSI interface board, Adaptec SCSI cable and a Kofax Adrenaline Capture Engine (ACE – without bar code & patch code support) or a Kofax Adrenaline Image Processing Engine (AIPE – with bar code & patch code support)
Bell & Howell 1000FB/5000F SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Bell & Howell 1000FB/5000F Video	Adrenaline 850 Video	n/a
Bell & Howell 1500FB/6000F SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Bell & Howell 1500FB/6000F Video	Adrenaline 850 Video	n/a
Bell & Howell 2000D FB SCSI	Adrenaline 850 SCSI	n/a
Bell & Howell 2135 Video	Adrenaline 850 Video	n/a
Bell & Howell 2137A Video	Adrenaline 850 Video	n/a
Bell & Howell 2138A Video	Adrenaline 1700 Video	n/a
Bell & Howell 3338/3338A Video	Adrenaline 850 Video	n/a
Bell & Howell 4040S/4040D SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Bell & Howell 6138 Video	Adrenaline 850 Video	n/a
Bell & Howell 6300 Video	Adrenaline 1700 Video	n/a
Bell & Howell 6338A Video	Adrenaline 1700 Video	n/a
Bell & Howell 8080/8125 SCSI	Adrenaline 1700 SCSI	ACE or AIPE 3.22.04
Bell & Howell 8080/8125 Video	Adrenaline 1700 Video	n/a
Bell & Howell PS100 Video	Adrenaline 1700 Video	n/a
Bell & Howell PS120V Video	Adrenaline 1700 Video	n/a
Bell & Howell PS150V Video	Adrenaline 1700 Video	n/a
Canon DR3020 SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Canon DR5020/5080 SCSI	Adrenaline 1700 SCSI	ACE or AIPE 3.22.04
Fujitsu M3091DC SCSI	n/a	ACE or AIPE - Coming Soon - 3.3
Fujitsu M3092DC SCSI	n/a	ACE or AIPE - Coming Soon - 3.3
Fujitsu M3093DE Video	Adrenaline 850 Video	n/a
Fujitsu M3093DG SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Fujitsu M3093E Video	Adrenaline 850 Video	n/a
Fujitsu M3093EX Video	Adrenaline 850 Video	n/a
Fujitsu M3093GX SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Fujitsu M3096E/E+ Video	Adrenaline 850 Video	n/a
Fujitsu M3096EX Video	Adrenaline 850 Video	n/a
Fujitsu M3096GX SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Fujitsu M3097 VRS Video	Adrenaline 850 Video	n/a
Fujitsu M3097DE Video	Adrenaline 850 Video	n/a
Fujitsu M3097DG SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Fujitsu M3097E+ Video	Adrenaline 850 Video	n/a
Fujitsu M3097GX SCSI	Adrenaline 850 SCSI	ACE OR AIPE 3.22.04
Fujitsu M3099A Video	Adrenaline 1700 Video	n/a
Fujitsu M3099EH Video	Adrenaline 1700 Video	n/a
Fujitsu M3099EX Video	Adrenaline 1700 Video	n/a
Fujitsu M3099G SCSI	Adrenaline 1700 SCSI	ACE or AIPE 3.22.04
Fujitsu M3099GH SCSI	Adrenaline 1700 SCSI	ACE or AIPE 3.22.04

Scanner	Kofax Adrenaline interface board, Adrenaline cable and Adrenaline software driver	Adaptec 1542 or 2940 SCSI interface board, Adaptec SCSI cable and a Kofax Adrenaline Capture Engine (ACE – without bar code & patch code support) or a Kofax Adrenaline Image Processing Engine (AIPE – with bar code & patch code support)
Fujitsu M3099GX SCSI	Adrenaline 1700 SCSI	ACE or AIPE 3.22.04
Fujitsu M4099D Video	Adrenaline 1700 Video	n/a
Fujitsu Scan Partner 93 SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Fujitsu SP jr. SCSI	n/a	ACE or AIPE 3.22.04
Fujitsu SP10c SCSI	n/a	AIPE 3.22.04
Fujitsu SP15C SCSI	n/a	ACE or AIPE 3.22.04
Fujitsu SP600c SCSI	n/a	ACE or AIPE 3.22.04
Hewlett-Packard 4c, 4p, 5p SCSI	n/a	ACE or AIPE 3.22.04
Hewlett-Packard 6250 SCSI	n/a	ACE or AIPE 3.22.04
Hewlett-Packard IIc, IIp, IIcx SCSI	n/a	ACE or AIPE 3.22.04
Hewlett-Packard IIIp SCSI	n/a	ACE or AIPE 3.22.04
Kodak ds1500 SCSI	Adrenaline 850 SCSI	n/a
Kodak ds2500 SCSI	Adrenaline 850 SCSI	n/a
Kodak ds3500 SCSI	Adrenaline 1700 SCSI	ACE or AIPE 3.22.04
Kodak ds3510 SCSI	Adrenaline 1700 SCSI	n/a
Kodak ds3590c SCSI	Adrenaline 1700 SCSI	n/a
Kodak ds5500 SCSI	Adrenaline 1700 SCSI	ACE or AIPE 3.22.04
Kodak ds7500 SCSI	Adrenaline 1700 SCSI	n/a
Kodak ds9500 SCSI	Adrenaline 1700 SCSI	n/a
Kodak IL500 SCSI	Adrenaline 1700 SCSI	n/a
Kodak IL900/923 SCSI	Adrenaline 1700 SCSI	n/a
Kodak IL990/ds990 SCSI	Adrenaline 1700 SCSI	n/a
Panasonic KV-S6040 SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Panasonic KV-S6045 SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Panasonic KV-SS2055 SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Panasonic KV-SS855 SCSI	Adrenaline 1700 SCSI	ACE or AIPE 3.22.04
Panasonic SS-25 SCSI	n/a	ACE or AIPE 3.22.04
Panasonic SS-50 SCSI	n/a	ACE or AIPE 3.22.04
Panasonic SS-55 SCSI	n/a	ACE or AIPE 3.22.04
Ricoh IS420 SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Ricoh IS420 Video	Adrenaline 850 Video	n/a
Ricoh IS430 SCSI	Adrenaline 850 SCSI	ACE or AIPE 3.22.04
Ricoh IS430 Video	Adrenaline 850 Video	n/a
Ricoh IS450 DE SCSI	Adrenaline 850 SCSI	n/a
Ricoh IS450 SE SCSI	Adrenaline 850 SCSI	n/a
Ricoh IS50 SCSI	n/a	ACE or AIPE 3.22.04
Ricoh IS60 SCSI	n/a	ACE or AIPE 3.22.04

Notes

- All scanners supported by Kofax's Adrenaline 850 interface board are also supported by Kofax's Adrenaline 1700 interface board.

- For information regarding scanners supported by Kofax's older KF series interface boards, refer to www.kofax.com and select the Scanner Configurator button at the bottom of the page.
- If the software driver has a version number, the related scanner is supported by that software driver version number or greater.
- *n/a* means this scanner is not supported by this combination of hardware and third-party software products.

Understanding Color Support

The default DPI setting in FormWare Scan for processing color images is 300 DPI, although FormWare can process color images between 200 DPI to 400 DPI. This DPI setting is dependant on the scanner you select for color image processing. Some scanners (e.g., the Kodak i820) can only scan color images at 200 DPI, while the Kodak i840 can process a 300 DPI or a 400 DPI color image. Captiva recommends the following scanners for high-speed scanning of color images up to a resolution of 400 DPI:

Kodak i840

- Certifications: CE Mark, CISPR 22 Class B, C-Tick, and CUL
- Certifications: TUV, Taiwan CNS 13438 Class A
- Certifications: UL-1950
- Document Handling: 1000 Sheet ADF
- Formats/Standards: JPG, CCITT Group III/IV uncompressed
- Maximum Scan Size: 12 x 30 in.
- Electronic Color Dropout
- Resolution: 400 DPI (optical) bitonal
- Scan speed: 160 PPM
- ISIS and TWAIN drivers included
- Color sheet-fed
- CCD Charge Coupled Device
- Fast wide SCSI2 interface

Panasonic KV-SS905C

- Full Color Duplex Production Scanning at 98 ppm / 174 ipm
- 600 DPI binary and 400 DPI color resolution
- Advanced Panasonic Image Enhancement Technology
- Dynamic Threshold Technology
- RTIV Image Capture Utility
- Selectable Black/White reference
- Multi-color drop-out
- Multi Stream® output
- ISIS® Certified and TWAIN Compliant drivers

Fujitsu fi-4990C

- Technology Color CCD (charge-coupled device) image sensor
- Output Resolution (DPI) 100, 150, 200, 240, 300, 4001 (user selectable)
- Optical Resolution (DPI) 400
- Halftone patterns Dither, error diffusion
- Scanning speeds letter/portrait simplex and duplex
- Monochrome 200 DPI 90 ppm 180 ipm
- Color 150 DPI 90 ppm2 180 ipm2
- Document feeding mode: Automatic Document Feeder (ADF)
- ADF capacity 1,000 sheets (selectable 250, 500)
- Industry-standard TWAIN and ISIS® drivers
- Minimum document size A7 (2.875 in. x 4.125 in.)
- Maximum size A3, double letter (11.7 in. x 17 in.)
- Ultra Wide SCSI interface and expansion slot
- Image type output binary; halftone; grayscale; color

Installation Scenarios

Three basic types of installations are supported by FormWare:

- **Suite Installer** – Installs a File Server (FormWare File System or FormWare Database) and FormWare stand-alone components.
- **Client Installer (NetSetup)** – Sets a computer up as a client workstation to a File Server. You must have DIRECT connectivity to the File Server to be able to use this installer, because a valid path to the File Server must be specified.
- **Remote Client Installer (Distributed Processing)** – Installs all files necessary for remote (i.e., Internet) client workstations to operate against a File Server when they communicate through a Web Server (FWCGI) and Internet FormWare Server (IFS) i.e., no direct connectivity to the File Server is required.

Installing the File Server

Follow these steps to install the FormWare File System software on the FormWare File Server. During installation, you can change an installation setting by selecting **Back** until you reach the previous window where the change is needed. Otherwise, select **Next** to continue to the next step. You can select **Cancel** at any time and then select **Yes** to stop the installation.

1. Insert the FormWare Installation CD in the CD-ROM drive. The installer program starts automatically and the FormWare Main Menu displays. If Windows AutoRun feature is disabled and fails to display the Main Menu, start the installer program by double-clicking on the Setup.exe program at the root level of the CD.

2. Select **Install FormWare v5.0 Application**. The Welcome window displays.

3. Select **Next** to display the License Agreement window.

4. Scroll through the license agreement and review the terms and conditions. If you agree with the terms, select **Yes** to continue. The Choose Installation Type window displays.

5. Select the **FormWare - File System** icon. The Choose Destination dialog displays.

6. Select **Next** to accept the default location (C:\FWShare\FormWare) or **Browse** to identify a new destination directory and then select **Next.** The Select Components dialog displays.

7. Select the components that apply to the current installation. Be sure to scroll to the end of the Select Components menu and select any appropriate recognition engine(s) and start-menu options. As you scroll down the list of components, notice that the **FormWare Security License Manager** (SLM) is selected by default. Select **FormWare for Panagon** only if Panagon is installed on this machine.

Caution: Although the installer permits you to install every component you select, the FormWare functions you can actually run in your forms processing application are restricted by the licenses you have purchased. If you check FormWare for Panagon, the setup program checks your machine for an installed version of Panagon, and if it is not installed, you will receive an error message.:

Note: The Start Menu Shortcuts option includes shortcuts that appear on your Windows **Start > Programs** menu. You might want to expand the Shortcut option. Every node showing a + sign can be expanded to reveal more components.

8. Select **Next**. The Select a FormWare Security License File dialog displays. You will typically select the SLM during most installations. However, if you did not select SLM on the Select Components dialog, the FormWare Security License File dialog will not display.

9. Select the **Browse** button to display the Select DAT File window and locate the path to the *.dat file with your license information. Insert the Captiva diskette containing the *.dat file or locate the file on your computer or the network. Select the *.DAT file and then **Open** to insert the .dat file path in the **SLM DAT File Source Location** field and then select Next.

Note: The diskette provided by Captiva contains the .dat file and a ReadMe file that may contain additional instructions for each installation. The *.dat file you select contains the licensing information about the FormWare components you purchased.

10. Select **Next**. The Select Program Folder dialog displays.

11. Select **Next** to add the programs, icons, or shortcuts to the default directory or type a new directory name in the **Program Directory** field and then select **Next**. The Start Copying Files dialog displays.

12. Review the Current Settings summary. Select **Back** to change any settings on previous dialogs or select **Next** to accept the selected settings and install the files. The Setup Status bar displays the progress of the installation.

13. When the FormWare installation is complete, the InstallShield Wizard Complete screen displays and asks you to reboot the machine. If you are using USB keys for SLM, delay the restart and continue with step 14.

14. If your FormWare implementation of SLM depends on a black HASP USB key to activate permissions to run then FormWare components, attach that key now at the USB port.

15. The Windows hardware wizard launches and auto-installs the drivers for the USB device.

16. When the installation is complete, you must restart your computer. Select **Finish** to complete application setup (recommended) or select **No** and select **Finish** to restart later.

17. On reboot of the server the License Manager starts.

After your computer restarts, you can begin using FormWare. You may also install the client workstation or the FormWare stand-alone components.

Note: The FormWare software does not run on a Novell Server. You can use a Novell server as a location to hold FormWare files (i.e., a file server), but it cannot run any processes. You can run the installer on a Windows machine to copy the FormWare tree to a Novell server. Run the Suite installation and point to a remote location for the directory where you want to install FormWare. For example, instead of installing to c:\FWShare\FormWare, install to \\<NovellServer>\FWShare\FormWare. The share drive must exist before you try to use it. The installer will copy all necessary files to the Novell Server, but not perform any actions on the machine from which you are installing because FormWare will not actually reside there.

Installing Client Workstations

Providing Shared Access to the Server

The FormWare File System files are installed on a File Server, and the client workstation is installed on a separate machine within the same network. Before client workstations can be installed, you must provide shared access to the FormWare directory on the File Server. To do this from Windows Explorer:

1. Right-click on the directory above the FormWare directory on the File Server.

2. Select **Sharing**.

3. Select the **Share this directory** option.

The files in the Netsetup directory will link the client workstation to the server. The Netsetup directory on the File Server was installed when you selected Client Installation (Netsetup) on the Select Components dialog during the server installation.

Caution: Running anti-virus software on FormWare client machines affects the performance of the FormWare application. Certain settings in your anti-virus software may make FormWare unusable. For example, if you are running Norton Anti-Virus and have the program set to Auto-Protect, Norton will check every file on the workstation as FormWare processing occurs. Norton recommends that you run in Auto-Protect mode unless you have been directed to disable it by a software package that you are installing.

Installing a Client Workstation

Follow these steps to install the client software across the network by pointing to the files in the NetSetup directory on the File Server rather than installing FormWare on the client workstations using the CD.

1. On the File Server, use Windows Explorer to go to the \\FWShare\FormWare\Netsetup directory. (\\FWShare\FormWare is the system-default path for FormWare. If you changed the default path during installation, locate the Netsetup directory somewhere in your environment.)

2. Double-click the **setup.exe** program to launch the client installer. The Welcome window displays.

3. Select **Next**. The Choose FormWare Directory Structure dialog displays.

---

4. Select **Next** to accept the default FormWare directory path. The default path results from having launched the client installation from the NetSetup directory. This default path will be sufficient for most environments. If you want to change the path, use the **Browse** button to select a valid path and then select **Next**.

5. On the Select Components dialog, all components installed on the File Server are checked by default. Leave only those components selected that you want installed on this client and deselect the rest.

6. Select **Next**. The Select Program Folder screen displays.

Note: Although the installer permits you to install every component you select, the FormWare functions you can actually run in your forms processing application are restricted by the licenses you have purchased.

---

7. Accept the default program directory or type a new directory name and select **Next**.

8. Review the Current Settings for the client workstation setup. Select **Back** to change specific settings on previous windows or **Next** to start copying the workstation client files. The Setup Status window displays the installation progress. When file copy is complete, the InstallShield Complete window displays.

9. (Optional) If this client will run InScript or InScript2, attach now the purple HASP USB key at the USB port on the client chassis.

10. (Optional) The Windows hardware wizard launches and recognizes the USB key. The wizard auto-installs the driver for the USB key.

11. Before you can use the client workstation for FormWare tasks, you must restart your computer. Select Finish to restart (recommended).

Installing an Unattended Client Workstation

The Client Installer includes unattended scripting functionality. It generates a script file (setup.iss) that reflects the choices a user makes through the initial installation dialogs. The script can be used to play back the installation on similarly configured client machines (i.e., same operating system).

Generating the Unattended Script File

Take the following steps to generate the unattended script file:

1. Run the client-install setup with a -r flag, which puts the setup into "record" mode (i.e., run the client installation on a machine using the "setup.exe -r" command line).

2. The installation launches as usual, except that it also creates the setup.iss file and places it in the Windows directory (WinNT directory on NT machines).

3. After the installation finishes, save setup.iss to the side for use later during the unattended installation. For example, copy setup.iss from c:\windows to c:\tmp\ISFiles.

Initiating an Unattended Installation

The installation is completely silent except for a single status dialog stating that the setup is running in unattended mode. The following procedure applies when setup.exe is in a writable directory:

1. Put setup.iss in the directory containing setup.exe and then run setup.exe with the -s flag.

---

2. With setup.iss in the same directory as setup.exe, the installer picks up setup.iss and uses it.

3. The installer generates the results of the unattended installation in a log file named setup.log that gets placed in the same directory as setup.exe.

4. Review the log file to see the success/error code results of the unattended installation.

The procedure changes when setup.exe is in a non-writable directory (i.e., on a CD). When calling setup.exe you must use flags for specifying the location of the .iss and .log file on the command line.

- Use the **-f1 flag** to allow specification of the .iss file.
- Use the **-f2 flag** to allow specification of the log file.
- Eliminate spaces between the argument and the subsequent path. For example

```
setup.exe -s -f1C:\tmp\setup.iss -f2C:\tmp\setup.log
```

Above example will launch setup in unattended playback mode using the unattended script file c:\tmp\setup.iss and will output the log results to c:\tmp\setup.log.

Installing the Security License Manager

The FormWare Security License Manager (SLM) communicates with each software component requesting security and verifies that a license for that component exists. When a user logs in, the total license count for that particular component is incremented. It is decremented only when the user logs out of the SLM.

Using the SLM, you can view the login information and available license information. The following procedure assumes that FormWare has been installed on the File Server on a network to which this standalone workstation is attached. You will only need to perform these steps if you have chosen not to run the SLM on your File Server (e.g., if your File Server is a Novell file server and cannot run SLM.)

Note: Make sure you are installing on a machine that does not have an existing File Server or FormWare client installation. This stand-alone component installation includes the software necessary for using the SLM.

Follow these steps to install SLM as a stand-alone option for use in the FormWare File System.

1. Insert the FormWare Installation CD in the CD-ROM drive. The installer program starts automatically and the FormWare Main Menu displays.

2. Select **Install FormWare v5.0 Application.** The Welcome window displays.

3. Select **Next** to display the license agreement.

4. Review the terms of the license and select **Yes** to continue. The Choose Installation Type dialog displays.

5. Select the **Stand-Alone Components** icon and select **Next.** The Choose Destination Location dialog displays.

6. Select **Next** to accept the default directory or select **Browse** to select a different destination directory ant then select **Next**. The Select Components dialog displays.

7. Select **FormWare Security License Manager** and select **Next**. The Choose FormWare Directory Structure dialog displays.

8. Select **Browse** to locate and select the FormWare directory on the server. (This step correlates the SLM running on this computer with the FormWare File Server located on the network.)

9. Select **Next.** The Select a FormWare Security License File dialog displays.

10. Select the **Browse** button to display the Select DAT File window and locate the path to the *.DAT file. Insert the Captiva diskette containing the *.DAT file or locate the file on your computer or the network. Select **Open** to insert the .DAT file path in the **SLM DAT File Source Location** field and select **Next**.

Note: The diskette provided by Captiva contains the .DAT file and a ReadMe file that may contain additional instructions for each installation. The *.DAT file you select contains the licensing information about the FormWare components you purchased.

---

11. On the Security License Manager Information window, select **Browse** to find a valid user name. At the Select User window select the row with the valid name and select **OK**. Then, select **Next** to validate the user name.

12. Review your setup on the Start Copying Files dialog and select **Next** to begin the installation. Setup installs the FormWare SLM and displays a Setup Status window and a progress bar.

13. When the installation is complete, select **Finish** and reboot your machine.

Installing QA Statistics

The QA Statistics module enables you to capture statistical data on errors that occur in Completion such as field-level and character-level errors made by either an operator or by OCR.

Capturing this type of statistical data is important because it differentiates between actual operator errors from OCR errors and enables management to track these statistics and tie pay increases or termination directly to operator performance. The QA Statistics module tracks field-level and character-level data at every stage within OCR and Completion and presents that raw data in format for reporting purposes.

The QA Statistics module can be implemented with the FormWare File System, FormWare Database, and FormWare for InputAccel.

Note: The QA Statistics module can only be installed on the FormWare server. The QA Statistics component will not appear in the "Stand-Alone" installation path for this reason. When installed as part of the FormWare File System, the QA Statistics module uses Microsoft SQL Server as its data repository. Before you install QA Statistics as part of your FormWare File System installation, verify that a SQL Server, which will hold the QA Statistics database schema, is installed on a server machine on the network. The installation steps in this section presume that SQL Server is already installed and requires the proper access to that area to populate the database with statistics information.

You will need the following for this installation:

- User with administrator rights to install FormWare
- FormWare Installation CD-ROM
- Diskette containing the FormWare license file (*.DAT) that contains the number of concurrent licenses you will need for the QA Statistics module (see note that follows)

- A hardware key (black HASP USB or blue HASP USB/Parallel key) required for the FormWare License Manager to license the software components (QA Statistics is a licensed component)

Note: The QA Statistics module requires a separate license to run within FormWare. The number of actual licenses you purchase depends on the number of simultaneous FormWare client workstations you have running that are using QA Statistics.

Follow these steps to install QA Statistics as part of a new FormWare File System installation.

1. Insert the FormWare Installation CD in the CD-ROM drive. The installer program starts automatically and the FormWare Main Menu displays. If Windows AutoRun feature is disabled and fails to display the Main Menu, start the installer program by double-clicking on the Setup.exe program at the root level of the CD.

2. Select **Install FormWare v5.0 Application**. The Welcome window displays.

3. Select **Next** to display the License Agreement window.

4. Scroll through the license agreement and review the terms and conditions. If you agree with the terms, select **Yes** to continue. The Choose Installation Type window displays.

5. Select the **FormWare - File System** icon. The Choose Destination dialog displays.

6. Select **Next** to accept the default location (C:\FWShare\FormWare) or **Browse** to identify a new destination directory and then select **Next**. The Select Components dialog displays.

7. Select the components that apply to the current installation. Be sure to check the **QAStats** option within the FormWare Application section. Scroll down to the end of the Select Components menu and select any appropriate recognition engine(s) and start-menu options.

Caution: The installer permits you to install every component you select; however, the FormWare functions you can actually run in your forms processing application are restricted by the licenses you have purchased.

Note: The Start Menu Shortcuts option includes shortcuts that appear on your Windows **Start > Programs** menu. You might want to expand the Shortcut option. Every node showing a + sign can be expanded to reveal more components.

8. Select **Next**. The Create QA Stats Database dialog displays.

9. Check the **Allow setup to create database** option if you want to create a database area on the SQL server for QA Statistics. If you leave this box unchecked, you must manually create this database yourself or it must already exist.

10. Select **Next**. The QAStats Database Configuration dialog displays.

Note: Once the installation process is complete, the name of the database server and database name are stored in the FWDB.INI file located in the \FormWare\Other folder. Each time you open the QA Statistics module, these values are verified against the file to ensure you have the authority to access this database.

11. In the **Database Server** field, type the name of the SQL server or select the **Browse** button to locate the SQL Server for the QA Statistics database. The SQL server must exist and be accessible from this workstation or the installation process cannot continue.

12. In the **Database Name** field, type the name of the database on the SQL server. This name must be unique. If it is not unique, a message will display indicating that this database name already exists.

13. In the **Database Username** field, type the SQL Server database system administrator user name.

14. In the **Password** and **Confirm Password** fields, type the password, if necessary.

15. Select the **Test** button. If connectivity succeeds, you will see the message "Configuration OK!" If database connectivity fails, setup displays an error message and details to help you pinpoint the source of the connectivity problem (e.g., Incorrect Client Software, Invalid User/Pass, or No Database Available).

16. Select **Next**. The Select a FormWare Security License File dialog displays. You will typically select the SLM during most installations. However, if you did not select SLM on the Select Components dialog, the FormWare Security License File dialog will not display.

17. Select the **Browse** button to display the Select DAT File window and locate the path to the *.dat file with your license information. Insert the Captiva diskette containing the *.dat file or locate the file on your computer or the network.

Note: The diskette provided by Captiva contains the .dat file and a ReadMe file that may contain additional instructions for each installation. The *.dat file you select contains the licensing information about the FormWare components you purchased.

18. Select the *.DAT file and then **Open** to insert the .dat file path in the **SLM DAT File Source Location** field and then select **Next**. The Security License Manager Information dialog displays.

19. In the **Username** field, type a valid domain user name or select the **Browse** button to display the Select User dialog. Select a user name with administrative rights to run the SLM Service.

20. In the **Password** field, type a valid domain password.

21. Press **[Tab]** to advance and retype the password in the **Confirm Password** field.

22. Select **Next** to authenticate the user account. The Select Program Folder dialog displays.

23. Select **Next** to accept the FormWare folder as the default and to have setup add program icons to that folder. The Start Copying Files dialog displays.

24. Review the **Current Settings** summary. Select **Back** to change any settings on previous dialogs or select **Next** to accept the selected settings and install the files. The Setup Status bar displays the progress of the installation.

25. If your FormWare implementation of SLM depends on a black HASP USB key to activate permissions to run the FormWare components, attach that key to the USB port. The Windows hardware wizard launches and auto-installs the drivers for the USB device.

26. Before you can use the FormWare programs and tools, you must restart your computer. Leave the **Yes** radio button selected and select **Finish** to complete the setup process and restart your computer.

Now that installation is complete, you can run the QA Statistics module within FormWare. See Setting Up QA Statistics in the Designing section.

Installing Scan Drivers

Using the FormWare Scan module you scan images that are then inserted into the JobFlow. You can also use Distributed Scan to scan forms at a remote location and send images, batches, and their associated files to the server. To do this you need to install a scan driver.

Installing ISIS Scanner Drivers

Before you select your scanner in FormWare Scan you must install the driver for your scanner:

1. On the scan client, insert the FormWare CD in the CD-ROM drive. The FormWare Scan Products installation options display.

2. Select the **Scanner Drivers** option. The Pick Scanner dialog displays the list of the scanners supported by FormWare Scan at the time of this release.

3. Scroll down the list and select your scanner.

Note: If you do not see your scanner in the list, FormWare Scan still may support your scanner. You can find out if FormWare Scan supports your scanner by going to www.scannerdrivers.com and viewing the scanner list. The level of your scanner must also match the level of FormWare Scan you have. A Level 3 scanner will not work with FormWare Scan Level 2.

4. Select **OK**. A message stating that your scanner driver was installed successfully displays.

Note: Be sure that your scanner is the scanner listed in the title bar of this dialog. If you picked the wrong scanner, close the dialog and go back to the beginning of this procedure.

5. Select **OK**. A message displays if it is necessary to reboot your system.

6. Select **Yes** to reboot. You will now be able to select the new scanner from within FormWare Scan.

Installing Kofax Scanner Drivers

Be sure you have exited all Windows programs before continuing.

1. Insert the FormWare Installation CD in the CD-ROM drive. The FormWare Main Menu displays.

2. Select **Install FormWare v5.0 Application.** The Welcome window displays.

3. Select **Next** to display the license agreement.

4. Review the terms of the license and select **Yes** to continue. The Choose Installation Type dialog displays.

5. Select the **Stand-Alone Components** icon and select **Next** to display the Choose Destination Location dialog.

6. Select **Next** to accept the default directory or select **Browse** to select a different destination directory and then select **Next**. The Select Components dialog displays.

7. Select **Install Scan Drivers**.

8. Select either the **KF-Series Runtime v3.2** or the **Adrenaline Runtime v3.6** and follow the screen prompts.

9. Select **Finish** to restart your computer.

Installing Adobe Acrobat Reader

FormWare user guides and release notes have been provided in PDF format on the FormWare CD. To view, print, and search these files, you will need to install the Adobe Acrobat Reader. If you have already installed Acrobat Reader on your computer, you can skip this procedure. Follow these steps to install Acrobat Reader:

1. Insert the FormWare Installation CD in the CD-ROM drive. The installer program starts automatically and the FormWare Main Menu displays.

2. Select **Install Adobe Acrobat v5**. The Adobe Reader Setup window displays.

3. Select **Next**. The Choose Destination dialog displays.

4. Accept the default location and select **Next** or select **Browse** to select a different destination and then select **Next** to begin copying Adobe files. The progress bar on the Setup Status window displays.

5. When files finish copying, select **Finish** to complete the Adobe Acrobat Reader installation.

Installing Components for Distributed Processing

This section describes the installation procedures you need to perform to establish Distributed Processing. You need to install the appropriate configurations for the servers, stand-alone components, and the remote workstations.

All of the Distributed Processing components require installation of a Web Server, an IFS, and a File Server at the host site. These three servers can reside on the same machine or can be split into two or three separate machines. However, you must plan how and where they will be installed before you begin the installation. Drawing a diagram of the intended installation with the name of each machine name will assist you in installing the different components successfully.
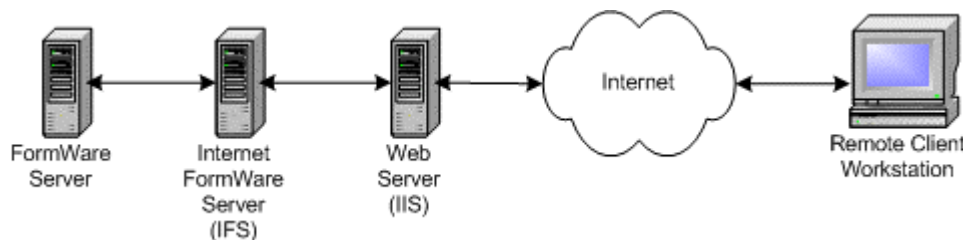
Setting Up the Servers

Set up the servers at the host site in the following order:

1. **File Server** - The File Server is the machine that runs the complete FormWare software suite. Successful File Server installation also includes the installation of the Security License Manager. The license file must be configured with an IFS license and the Security License Manager service must be running to run any of the Distributed Processing components.

2. **Internet FormWare Server (IFS)** – The IFS prepares data for submission into a JobFlow and acts as the liaison between the Web Server and the File Server. During the IFS installation, a minimal set of files necessary to run the communication service between the Web Server and the File Server are installed. This component can be installed on the File

Server machine or on a separate machine. If you install this service on a separate machine, that machine must be able to communicate with the File Server over the LAN. This service requires user configuration after installation and the FormWare Internet Server Service must be running before Distributed Processing can occur.

3. **Web Server with the FWCGI component** – The Web Server is the host site's "front end" link to the Internet. This server is where your company web site and pages reside. Before installing the FWCGI component, you must already have a Microsoft IIS Web Server installed and configured to allow remote users to successfully connect to the Web Server over the Internet. The FWCGI installation copies a minimal set of files to the Web Server, including a configuration file that references the machine name of an IFS.

Caution: When installing the IFS and the FormWare Common Gateway Interface (FWCGI), make sure you install these stand-alone components on a machine that *does not* have an existing File Server or FormWare client installation. The stand-alone component installation includes the software necessary for running the component.

Installing the IFS

The IFS prepares data for submission into a JobFlow and is the liaison between the Web Server and the File Server. You install IFS as a service on Windows NT machines, but you do not need to run it as a service. Since the IFS is installed as a service, the name FormWare Internet Server displays in the Service Manager.

All communication between remote clients and the IFS is routed via the HTTP protocol through the web server to the IFS. You do not need to install the IFS component on the File Server; it can be installed on another machine in your FormWare network. A parameter in the ICapture.ini file points to a valid IFS Host in the FormWare environment. The IFS installation configures the INI files on the File Server. The installation process does the following:

- The FWCGI installation writes IFS Host entries to the FWCGI.ini file.
- The IFS install writes the IFS Host entries to the ICapture.ini file.
- The IFS installation ensures that the LFN=Y parameter exists in the System.ini file. Setup also ensures that the YEAR=4DIGIT parameter exists in the System.ini file.
- The IFS installation ensures that the port IFS uses is properly entered into the \System32\Drivers\Etc\Services file (i.e., Icapture 10098/tcp).

Perform the following steps to install the IFS:

1. Insert the FormWare Installation CD in the CD-ROM drive. The installer program starts automatically and the FormWare Main Menu displays.

2. Select **Install FormWare v4.6 Application.** The Welcome window displays.

3. Select **Next** to display the license agreement.

4. Review the terms of the license and select **Yes** to continue. The Choose Installation Type dialog displays.

5. Select the **Stand-Alone Components** icon and select **Next** to display the Choose Destination Location dialog.

6. Select **Next** to accept the default directory or select **Browse** to select a different destination directory ant then select **Next**. The Select Components dialog displays.

7. Select **IFS** and then **Next**. The Choose FormWare Directory Structure dialog displays.

8. Select **Browse** to select a valid FormWare directory and then select **Next.** The IFS Service Information dialog displays.

9. Type a valid user name or select **Browse** and select a valid name.

10. Type and confirm the password, if applicable. Then, select **Next**. The Start Copying Files dialog displays.

11. Select **Next** to display the progress bar on the Setup Status window.

12. When files finish copying, select **Finish** to complete the IFS installation.

Installing the FWCGI

Captiva recommends that you install the FormWare Common Gateway Interface (FWCGI) on an established Web Server. The Web Server uses CGI to handle dynamic content. When a request is made from the Web Server to FWCGI, FWCGI dynamically generates the content and sends it via the Web Server to the client browser.

During installation the FWCGI.exe and FWCGI.dll files are copied to the CGI-BIN directory on the Web Server. This directory must include Execute permissions for communication to be successful with the FormWare Internet Server service. Use the Web Server's Management Configuration utility to assign the correct permissions to this directory.

The installation process also puts the FWCGI.ini file in the FWShare\FormWare\INI\System directory on the Web Server and adds the FWCGI_CNFG system environment variable to the Web Server system environment. This variable points to the location of the configuration file.

To install FWCGI, you must have the following:

- File Server
- IFS
- HTTP/1.1-compliant web server (Apache or IIS). (IIS is the more common web server on Windows platforms.)
- Web browser such as MS Internet Explorer or Netscape
- TCP/IP network connectivity (not necessarily Internet)
- Appropriate licenses and hardware key

The following sections provide instructions for installing the Distributed Processing components from the FormWare CD. You can also install them from an HTML page that you access with a browser. The FormWare CD has a sample editable installation HTML page.

To install the FWCGI, follow these steps:

1. Insert the FormWare Installation CD in the CD-ROM drive. The installer program starts automatically and the FormWare Main Menu displays.

2. Select **Install FormWare v4.6 Application.** The Welcome window displays.

3. Select **Next** to display the license agreement.

4. Review the terms of the license and select **Yes** to continue. The Choose Installation Type dialog displays.

5. Select the **Stand-Alone Components** icon and select **Next** to display the Choose Destination Location dialog.

6. Select **Next** to accept the default directory or select **Browse** to select a different destination directory ant then select **Next**. The Select Components dialog displays.

7. Select **FWCGI** and then **Next.** The Specify IFS Host dialog displays.

8. Select **Browse** and select the name of a valid IFS host or you can type a valid name in the **IFS Host** field. Then, select **Next.** The Choose location of the web root directory dialog displays.

9. Select **Next** to accept the default location or select **Browse** to select a different path to the web root directory and select **Next.** The Start Copying Files dialog displays.

10. Review your FWCGI settings. If the settings are correct, select **Next** to install FWCGI and display the progress bar on the **Setup Status** window.

11. When files finish copying, select **Finish** to complete the installation.

Installing on Remote Workstations

The FormWare remote components handle all of the communication to and from the Web Server. The remote client uses TCP/IP networking to transfer data to and from the IFS. Because it goes through a Web Server, the remote client workstation needs no knowledge of the File Server domain. Therefore, the domain structure typical in a LAN-based FormWare environment need not be in place for a remote client.

The Distributed Processing (Remote) installation program installs all components necessary to run the FormWare software on a remote workstation. It creates the FormWare directory structure, sets the environment variables, sets the path statements, and creates icons within the Start menu.

You can install the remote workstation Distributed Processing components from the FormWare CD, a shared FormWare network directory, or from an HTML page accessed from the remote workstation.

Note: Exit all applications and stop all background programs, such as virus protection, before installing any FormWare application.

Installing Remote Components from the FormWare CD

To save Internet download time, you can select setup.exe in the Remote directory on the FormWare CD to install the files needed for remote processing.

Note: FWShare and FormWare represent system-assigned names in the path for launching setup.exe. In your installation, these would be whatever names you designated when you installed the FormWare software on the File Server.

To install remote components from the FormWare CD:

1. Insert the FormWare CD into the remote workstation CD drive. The installation program opens automatically.

2. Close the installation program; you access the remote installer from another program.

3. Select **Run** from the Windows Start menu.

4. Select **Browse** and locate the setup.exe program in the **Remote** folder on the CD.

5. Select **OK** and double-click setup.exe to start the wizard that guides you through the installation process.

6. A message ("FormWare Client Setup is preparing . . . ") displays, followed by the InstallShield Welcome window for FormWare v4.6 Remote.

7. Select **Next** to display the Choose Destination Location dialog.

8. Accept the default folder (C:\FormWare) or browse for an alternative folder and insert it by selecting it and pressing **OK**.

9. Select **Next** to display the Select Components dialog.

10. Select a component. Options are:

    *   Distributed Completion

    *   Distributed Scan

    *   FormWare Filler

    *   Remote Transfer Utilities

    *   Security License Manager (SLM)

11. Select **Next** to display the Select Program Folder dialog.

12. The default selection is **FormWare Remote**. To select a different folder, refer to the list of existing folders or type the name of a new folder. Otherwise select **Next** to accept the default and to display the Start Copying Files dialog.

13. As files install, interim progress messages appear on the status screen.

14. When file copy is complete, the message, "File copy is 100%," displays, followed by the InstallShield Wizard Complete dialog.

15. The installation is complete, but before you can use the FormWare programs and tools on the remote workstation, you must restart your computer. Select **Finish** (recommended) to restart or "No" to restart later.

Installing Remote Components from the Web Server

The complete remote installation and a sample HTML page are also included on the FormWare installation CD. You may provide the HTML page installation to remote users by copying the contents of the installation CD's \Remote directory to your Web Server and adding a link to a Web page. An ActiveX control is installed on the client machine to handle downloading and installation. The speed with which the remote components install across the Internet is a function of the connection speed.

Caution: In a Distributed Processing deployment, it is important that the directory containing the Remote installation package is not granted "Execute" permissions within the directory's properties on the Web server hosting the Remote installer. If the "Execute" permission on the Remote directory is enabled, it could cause unusual slowness on the client. When using the Remote installer from a Web server, it is extremely important that all security updates from your Web server vendor are applied.

To install the remote components from your Web Server:

1. Connect to your host site URL using a Web browser to display the remote installation page (e.g., http:\\www.companyname.com\FWSetup.html).

2. Select the hyperlink to start the wizard that guides you through the installation process. The installation program displays a security window.

3. Select the option to trust the source and select **Next**. The Destination Location dialog displays.

4. Use the default directory path or select **Browse** to select an alternate path for the remote installation. Select **Next**. The Select Components window displays.

5. Select the check box next to the name of each component that you want to install. As you select a component, its description displays in the **Description** field. Below the component list are disk space monitors. The Space Required size increments as you select a component for installation. The available components available are:

- **Distributed Completion** - Installs Completion, VBA support, and the necessary remote transfer utilities.

- **Distributed Scan** - Installs Scan Operator, Scan Designer, and the necessary remote transfer utilities. The system also selects the Security License Manager required to run Scan Operator. Any remote client running Distributed Scan must be running the FormWare License Manager with the appropriate hardware key and license .DAT file installed.

- **FormWare Filler** - Installs FormWare Filler and the necessary remote transfer utilities. You can optionally install VBA support by selecting the VBA option under FormWare Filler.

- **Remote Transfer Utilities** - Installs the remote transfer utilities only. This option is primarily for transferring XML files, but you can also use it to transfer Batch Definition files (BDFs) in an ad-hoc fashion.

- **Security License Manager (SLM)** - Installs the License Manager and License Monitor programs. This option is only required for use with Distributed Scan (Scan Operator).

6. Select **Next** to display the review window where you verify the settings the program uses to start copying files.

7. Verify that all the setting are correct and select **Next** to start the installation. Select **Back** if you need to adjust any of the settings. Installation takes a few minutes.

8. Select **Finish** to restart your computer. Failure to restart your computer at this time may cause some applications to not run properly.

Once installed, the remote client workstation can begin performing tasks after it connects to the File Server. See the Connecting a Remote Client to a File Server section for more information.

Installing Distributed Processing Patches

Periodically, Captiva releases software patches that fix software defects and enhance the functionality of the product. Patches include release notes that give instructions on successfully installing into a FormWare environment. It is important to follow the release note instructions when installing a patch for a FormWare Distributed Processing component and environment. Patches can include updates to the files located on the Web Server, IFS, File Server, and remote workstations. If you do not install the correct patch files on all of the machines running these components, Distributed Processing will not work correctly.

Installing the Support Utilities

When you the install FormWare File System software, the FormWare setup automatically installs the support utilities for this system. Some of the support utilities are installed only with FormWare Database.

- MS DCOM95
- HASP key drivers
- MS XML Parser

- MSMQ
- MDAC
- VBA for FormWare

These utilities are third-party products. If you need to install them independently of the FormWare installer, refer to the installation instructions published by the respective third party. These utilities are provided as a separate installation on the FormWare CD in case a utility fails to install automatically during the Autorun setup.

Caution: If a utility fails to install during the normal installation process, we recommend that you contact Technical Support for assistance in installing one or more of these utilities.

Modifying Your Installation

From the FormWare CD, you can access the maintenance mode that assists you in modifying, repairing, and removing your FormWare installation. You can also access the maintenance mode options through the Add/Remove Programs option in the Control Panel on your computer.

Modifying the System

The setup program enables you to add new program components or select currently installed components to remove. On the computer where the FormWare components are installed do the following:

1. Navigate from the Windows Start button to the Add/Remove Programs feature (i.e., **Start > Settings > Control Panel > Add/Remove Programs**).

2. From the list of installed programs select **FormWare v4.6**.

3. Select the **Change/Remove** button to launch the Install Wizard. The Modify-Repair Welcome screen displays.

4. Select **Modify** and then **Next**. The Modify Installation Type screen displays.

5. Select **FormWare File System**. The Select Components dialog displays.

6. Choose the components you want setup to install and select **Next**. The Start Coping Files screen displays.

7. Select **Next** to copy the necessary files. The Setup Status screen displays, followed by the Maintenance Complete screen.

8. Select **Finish** to restart your computer and enable the new components.

Repairing the System

The Repair function enables you to reinstall all program components installed at previous steps (i.e., replace any missing files and restore your environment variables to their initially installed state). On the computer where the FormWare components are installed do the following:

1. Navigate from the Windows Start button to the Add/Remove Programs feature (i.e., **Start > Settings > Control Panel > Add/Remove Programs**).

2. From the list of installed programs select **FormWare v4.6**.

3. Select the **Change/Remove** button to launch the Install Wizard. The Modify-Repair Welcome screen displays.

4. Select **Repair** and then **Next**. The Perform Installation Repair message displays.

5. Select **Yes** to replace any missing files and restore your environment variables to their initially installed state.

6. The Setup Status screen displays, followed by the Maintenance Complete screen.

7. Select **Finish** to restart your computer and implement the FormWare repair.

Removing the System

The Remove option is a typical Windows un-install procedure with screens that walk you through the appropriate steps. Take the following steps to remove installed FormWare components:

1. Select the Windows Start button and then Settings > Control Panel > Add/Remove Programs.

2. From the "Currently installed programs:" list select the component you wish to remove.

3. Select **Change/Remove**.

4. At the Confirm Uninstall window select **OK** to remove all components of the selected application.

Establishing Security

Creating a User INI File

User INI files contain the parameters used during Completion for individual operators. The parameters within a user INI file enable to you to programmatically control how a particular user can gain access to or function within FormWare components. Each user INI file must have the unique name of the user on the system (.ini extension, no periods, and no spaces). This name is what users type when they log onto the system. To create a user INI file:

1. From the Administration window, open the \FormWare\INI\User folder.

2. Right-click in the Contents area and select **New** from the context menu. A text editor displays.

3. Type the desired parameters in the file. See INI Parameters in the Supplementary Information section for a list of the available parameters you can use within user INI files.

4. Select **Save** on the toolbar. The Save As dialog displays.

5. In the **File name** field, type the name of the user. This name must match the user's name on the system.

6. Select **Save** to save the file.

7. Select **File > Exit** to exit the text editor.

Creating a Group INI File

A group INI file contains a collection of FormWare users to whom you want to assign certain access and user rights. This allows you to control or limit the access to certain programs and functions within FormWare to only those users within a particular group. To have group rights, a user must be assigned to that group. This is accomplished by adding the user's login ID to the group INI file. Thereafter, a user logging onto Windows and FormWare with the correct login ID can access the areas of FormWare as defined by his or her group profile. To create a group INI file:

1. From the Administration window, open the \FormWare\INI\Group folder.

2. Right-click in the Contents area and select **New** from the context menu. A text editor displays.

3. Type the desired parameters in the file. See INI Parameters in the Supplementary Information section for a list of the parameters you can use within group INI files.

4. Select **Save** on the toolbar. The Save As dialog displays.

5. In the **File name** field, type the name of the group. This name must match the group's name on the system.

6. Select **Save** to save the file.

7. Select **File > Exit** to exit the text editor.

Establishing Security for Distributed Processing

Forcing Remote Operators to Login

The very first time remote workstation operators log on to a Distributed Completion session, they are required to type their user name (ID) and password to gain access to the session. This information is then stored in a RmtXfer.ini file on their workstation (the password is encrypted). Because this information is saved on the remote workstation, the operator is not required to log on to the host site for any subsequent Distributed Completion sessions.

By setting the RMT_FORCE_LOGIN parameter (RMT_FORCE_LOGIN=Y) in either the user or group INI file, you can force remote workstation operators to log in with a valid user name and password each time they access a Distributed Completion session. If the RMT_FORCE_LOGIN parameter is not set (or set to RMT_FORCE_LOGIN=N), Distributed Completion will continue to use the user name and password information in the RmtXfer.ini file and not require the user to provide it at startup.

See the Modifying User or Group INI Files for Distributed Processing section for instructions on setting this parameter.

Changing DAT Files During FWSLM Operation

You can change information in the license DAT file while the FormWare Security License Manager (FWSLM) is running. FWSLM dynamically detects if the DAT file has been changed and adjusts the internal license information accordingly. A component that is actively logged in using FWSLM is not affected unless the license count for that component decreases in the new DAT file. In this case, the component will fail the next time it performs its background security check, which occurs approximately every two minutes.

Specifying Recognition Speed at Login

Using FWSLM, you can specify the recognition speed a specific license during login. This allows you to control which license is activated first if two licenses are activated simultaneously. To set the recognition speed for a specific license add the FWSLM_OPTION=nnn parameter to the appropriate recognition engine INI file (e.g., the RSP.ini file for the RecoStar Pro recognition engine). Recognition engine INI files are located in the \FormWare\INI\System folder. The value of nnn is the recognition speed and must be a number. The speed must exist in the DAT file and an available license at that speed must be present. If not, the login will be denied.

Configuring the File Server for Distributed Processing

FormWare Distributed Processing uses a FormWare application created on the File Server. This application includes the necessary job, forms, edits, and other files necessary to process work through a FormWare JobFlow. Application files are synchronized to a remote client through a process called Remote Sync.

Understanding Remote Sync

The Remote Sync process synchronizes application files on the remote workstation with application files on the File Server for the distributed process assigned to the remote workstation. When you update a SCG, CFG, or JDF file on the server, all remote operators that use those files can obtain the updated version through the Remote Sync process. Remote Sync examines the user or group INI files on the File Server to determine which application files apply to a specific remote user. When a remote user logs in with a specific user name and password, Remote Sync uses that information to locate the corresponding user or group INI file. Remote Sync then synchronizes (copies) the files listed in the INI file (including the INI file) with those on the remote user's machine.

Performing the Remote Sync process should be a regular part of a remote user's activities. On subsequent Remote Syncs, the process only copies updated files to the user's machine. The time required to execute the Remote Sync process is a function of the speed of the connection and the size of the updated files.

Forcing the Remote Sync Process on a Remote Workstation

The first time remote workstation operators log on to a Distributed Completion session, the Remote Sync process is automatically launched on their remote workstation. The Remote Sync process verifies that the files on the remote workstation are current and match those on the File Server. However, during subsequent remote log ins, the remote operator must select the Remote Sync button to launch the Remote Sync process.

By setting the RMT_FORCE_REMOTE_SYNC parameter (RMT_FORCE_REMOTE_SYNC=Y) in the user INI file, you can force the Remote Sync process to occur every time the operators log in to a Distributed Completion session. If the RMT_FORCE_REMOTE_SYNC parameter is not set (or set to RMT_FORCE_REMOTE_SYNC=N), the remote operator will need to select the Remote Sync button to launch Remote Sync on subsequent log ins.

> **Note:** This parameter can only be set in the user INI file; it is not available for use the group INI file.

See the Modifying User or Group INI Files for Distributed Processing section for instructions on setting this parameter.

Modifying User or Group INI Files for Distributed Processing

You need a user or group INI file for each remote user to perform distributed processes. You create INI files on the File Server and transfer them to remote clients through the Remote Sync process. This ensures that the parameters contained within user and group INI files on remote workstations are always in sync with the "master" copy of that INI file on the File Server. Although the system employs user INI files to specify many user-specific settings, only the properties particular to distributed processing are described below.

Use the Remote_User_Sample.ini file in the \FormWare\INI\User directory as a template for your remote <user name>.ini files.

Three parameters in this file are required:

- RMT_HOSTNAME=<webserver name>
- RMT_USERNAME=<user name>
- RMT_PASSWORD=<password>

Change the parameter values (webserver name, user name, password) to valid values for this remote user's configuration.

To create a user INI file with Distributed Processing parameters:

1. On the File Server machine, start **Administration** and display to the \FormWare\INI\User folder in the File Browser.

2. Double-click on the Remote_User_Sample.ini file. The template INI file displays in a text editor.

3. Rename this file with a valid remote user name.

4. Replace the **<webserver name>** value for the **RMT_HOSTNAME** parameter with the name of your Web Server.

5. Replace the **<username>** value for the **RMT_USERNAME** parameter to the name of the remote user.

6. Replace the **<password>** value for the **RMT_PASSWORD** parameter to the remote user's password.

7. Change any of the remaining (optional) parameters as needed for this remote user's configuration. Remove the comment character (#) before each parameter you want to activate.

8. When you are finished configuring the file, select **File > Save As** to display the Save As dialog.

9. In the **File name** field, type the name of the remote user to which this file pertains and select **Save**. This name must match the name used for the remote user on the system.

10. **Save** your changes and **Exit** the file.

When the remote user logs in to the system, this INI file is copied to that user's remote workstation during the Remote Sync process. By maintaining a "master" copy of the user INI files for all remote users on your system, you can programmatically control their access to and ability to process work within the FormWare Distributed Processing system. Use the Remote_User_Sample.ini file template in the \FormWare\INI\User folder to create user or group INI files for all your remote users.

The following section describes the syntax and usage for the Distributed Processing parameters that you can put in a user or group INI file.

ALLOWJOBFLOW=<JobFlowName: Task#,Task#>	Specify the WorkFlow names and step numbers to which the remote user should be restricted. Use multiple ALLOWJOBFLOW parameters to allow access to more than one WorkFlow. Although this parameter is not specific to remote users, it is recommended practice to restrict the tasks to which a Distributed Completion user has access. For example: ALLOWJOBFLOW=CP_WORK:4,12
	Where CP_WORK is a WorkFlow name, and 4 and 12 are the accessible step numbers within the WorkFlow.
RMT_ALLOWBATCHSELECT=Y/N	Set this value to enable remote users to select a specific batch to process by transaction name. When RMT_ALLOWBATCHSELECT is set to Y, a Select Transaction dialog displays to the remote operator when he logs on to the system. The operator can type the transaction name or a template name to locate. The system will then pull the selected batch, or the first batch that matches the template entry. The remote user must type enough of the name of the transaction for the system to uniquely identify it. Wildcard characters (*) can be used for a template match.
	For example, if the desired transaction is represented in the WorkFlow as c:\formware\files\00001.bdf, the remote user would type "00001". The system will supply a .bdf extension when no extension is present, and will then look for the first batch to match the template *\00001.bdf. If it is possible to have BDFs with the same name in different subdirectories, the user must enter enough of the path to uniquely identify the batch. For example, if the desired transaction is represented in the WorkFlow as c:\formware\files\Monday\00001.bdf, the user must type "Monday\00001". The user can select **Cancel** to return to the main Remote Processing dialog, without selecting a batch. If the requested batch is not available, the user will receive an error message.
RMT_BATCHCOUNT=<n>	Set this value to the number of batches or data items that should be downloaded to a user's local machine during Distributed Completion. The default is 10.

RMT_EXCLUDE_FILES={<WorkFlow name>= {<step number>=<a>}}	Set this parameter to programmatically control the types of files that are downloaded to remote workstations during Distributed Completion. Files are excluded based on WorkFlow, step number, and file type. The following file types can be excluded from the download:
	
	D or d – Exclude transfer of batch data files
	
	I or i – Exclude transfer of primary image files
	
	O or o – Exclude transfer of OCR files
	
	E or e – Exclude transfer of enhanced image files
	
	T or t – Exclude transfer of thumbnail image files
	
	C or c – Exclude transfer of color image files
	
	For example in the John.ini file:
	
	RMT_EXCLUDE_FILES=
	
	{
	
	Test=
	
	{
	
	1=oetc
	
	}
	
	}
	
	When John checks work out of the Test WorkFlow, step one, OCR files, enhanced images, thumbnail images, and color images will be excluded from the download.

RMT_FILE_LOCATION= <FORMWAREPATH>	Set the path to the client workstation directory where you store batch files to be transferred. This is usually the same directory to which you scan. This property is only used for Distributed Scan. The following example indicates that if Remote Transfer is running the Remote Scan option, it should monitor the specified location for incoming BDFs and transfer them, and their corresponding image files, to the host site. RMT_FILE_LOCATION=d:\FormWare\Files\Send
RMT_FORCE_LOGIN=Y/N	Set this value to Y (Yes) to force remote workstation operators to retype their user name and password at the start of each Distributed Completion session. The default setting is N (No).
RMT_FORCE_REMOTE_SYNC= Y/N	Set this value to Y (Yes) to force the Remote Sync process to run every time a remote operator logs into a Distributed Completion session. If the RMT_FORCE_REMOTE_SYNC parameter is not set (or set to RMT_FORCE_REMOTE_SYNC=N), the remote operator must select the Remote Sync button to launch Remote Sync on subsequent log ins. **Note:** This parameter can only be set in the user INI file; it is not available for use in the group INI file.
RMT_HOSTNAME=<URL>	Specify the URL of the Web Server to which you will be connecting. The HTTP does not need to be specified and the www is optional. For example: RMT_HOSTNAME=www.companyname.com

RMT_JOBS=<JobName.jdf>	Set this value to the names of the jobs (JDFs) that should be copied to a remote workstation during a Remote Sync process. Forms (FDFs) and other job dependent files are not listed. The Remote Sync process uses the job name to determine dependencies. All other files, such as forms and check digits, are part of the job properties and are derived from the job name. Use the RMT_USERFILE parameter to specify any other files that need to be transferred to the remote workstation.
	Use an asterisk (*) as a wild card character within the job name. Separate each job name with a semi-colon. For example: RMT_JOBS=test.jdf;protect.jdf
	Following is a list of file dependencies automatically copied to a remote workstation during Remote Sync when you specify a JDF.
	File dependencies for each job:
	Form ID Training file – <JobName>.trn
	Dynamic ID file – <JobName>.id
	Job INI file – <JobName>.ini
	Manual Form ID file – <JobName>.idm
	Job Visual Basic Project file – <JobName>.vbp
	Job-specific Help File – <JobName>.hlp
	Reject Menu file – <JobName>.rej
	Field Edit runtime file
	File Edit runtime file
	Keyboard map file
	Export program runtime file
	Output device file
	Codeset file
	Auxdup file
	I-Forms script VBS file
	I-Forms script JVS file
	File dependencies for each form:
	Rule files, and all recursively nested rule files
	Background bitmap file
	Setup image file
	Overlay image file
	Removal image file
	File dependencies for each field:
	Field bitmap file
	Field list file
	Chkdigit file
RMT_OVERWRITE=T/F	Set this value to T (True) or F (False) to determine whether to write over files of the same name on the destination server with those being transferred. This option does not apply to Distributed Completion. For Distributed Completion, the overwrite condition is always set to T (true). For example: RMT_OVERWRITE=T

RMT_PASSWORD=<Password>	Type the password associated with the account. This does not correspond to any operating system password because it is only known and authenticated by the FormWare system. For example: RMT_PASSWORD=73greth
RMT_PREFETCHCOUNT=<n>	Set the number of batches to transfer before Completion is launched. The RMT_PREFETCHCOUNT must be less than the RMT_BATCHCOUNT. For example, if the RMT_BATCHCOUNT=10 and the RMT_PREFETCHCOUNT=2, the remote user will be able to begin processing after two batches have been downloaded, and files will be continuously transferred in the background.
RMT_PROXYPORT=<PortNumber>	Type the port number for the Proxy Server that will be used when a remote client connects to the host site. This setting is used if the RMT_USEPROXY parameter is set to Y. **Note**: The RMT_HOSTNAME, RMT_USERNAME, and RMT_PASSWORD parameters are required in the user INI file for Distributed Scan, FormWare Filler, XML Import, and Distributed Completion.
RMT_PROXYSERVER= <ProxyServerIPAddress>	Type the computer name or IP Address of the Proxy Server that will be used when a remote client connects to the host site. This setting is used if the RMT_USEPROXY parameter is set to Y.
RMT_RMTCFG=<Filename.cfg>	Specify the remote configuration file (CFG) to send with FormWare Filler. Using an asterisk, transfers all of the CFG files in the \FormWare\Remote\Configs folder from the central location to the remote workstation. For example: RMT_RMTCFG=job76.cfg
RMT_SCANCFG=<Filename.scg>	Specify the Scan Configuration files (SCG) to send with Distributed Scan. Using an asterisk transfers all of the SCG files in the \FormWare\Scan folder from the FormWare File Server to the remote workstation. For example: RMT_SCANCFG=scanop153.scg
RMT_SEARCH_SUBDIR=T/F	Set this value to T (True) or F (False) depending on whether you want subdirectories in the file location directory to also be searched for files to transfer. Use this property with RMT_FILE_LOCATION. The default is F. For example: RMT_SEARCH_SUBDIR=T
RMT_USEPROXY=Y/N	Set this value to Y if the remote client will connect to the host site via a Proxy Server. Set this value to N to disallow the capability.
RMT_USERFILE= <SUBDIRECTORY\FILENAME>	Specify any non-job related file to be transferred to the remote user machine. All files specified with this parameter must exist within the FormWare directory structure and are referred to by both the sub-directory name and filename. You can use an asterisk (*) as a wild card within the filename. You can use multiple lines to include files from multiple directories. The following example indicates the system should copy all PRP files in the \FormWare\Rules directory to the remote user's machine during Remote Sync: RMT_USERFILE=RULES*.PRP
RMT_USERNAME=<UserName>	Type the user name for a remote user's account (i.e., the user INI file name). It does not need to correspond to any operating system user name because it is only known and authenticated by the FormWare system. For example: RMT_USERNAME=oper216

RMT_WORKONLINE=Y/N	Set this value to Y to allow the user to begin work while batches are transferred to and from the server, completely in the background. This assumes that user has access to the server via a permanent and reliable network connection to the server (i.e., LAN connection and not a dial up connection).

Using the Operator Statistics File

Distributed Completion tracks operator statistics when remote Completion operators finish processing work. This statistical information is collected in the Operator Statistics file in the \FormWare\Statistics folder on the remote workstation and is transferred back to the File Server with the completed work. For the File Server to successfully receive this information, an Operator Statistics file must exist on the server in the same format as the remote workstation. All remote workstation installations are installed (by default) with a System.ini file containing the following settings:

- **LFN=Y** – Permits the storage and transmission of long filenames
- **YEAR=4DIGIT** – Converts all dates to use a four digit year (MMDDYYYY).

These settings govern how operator statistical data is stored on disk. The System.ini file on the File Server must also have these identical settings or transmission of operator statistic data will not be successful.

If the Operator Statistics file is ever deleted on the remote workstation, a new file must be generated. The remote Completion operator will receive a message indicating that the file was deleted and the File Server administrator will also receive a corresponding message. However, remote operators can continue to work without the file until the situation is resolved.

Configuring the IFS

The Internet FormWare Server (IFS) acts as a liaison between the Web Server and the File Server. During processing, the IFS must be running as a service (i.e., FormWare Internet Server Service) to accept and interpret incoming communications. All communication between remote workstations and the File Server is routed over the Internet, via the HTTP protocol, through the Web Server and IFS.

See the diagram in the Installing Components for Distributed Processing section for an illustration of this server relationship.

The IFS directory contains the following subdirectories:

- **Data** - Contains IFD files (I-Forms Data files), which are transactions submitted from I-Forms processing. IFD files are translated into BDFs and batch data files by the I-Forms Server for continued processing within JobFlow. Incomplete or failed open transactions also reside in this folder.
- **Logs** - Contains IFL files (I-Forms Log files) that collect and organize information about I-Forms transactions. Each I-Forms Server has its own log file, named according to the IFS. Double-click on a log file to view transaction information within an I-Capture Log viewer, including job name, submit type, transaction ID, date and time information, and other information that helps track I-Forms transactions.
- **Scripts** - Contains VBS files for Visual Basic scripts. The IFS uses these script files to run I-Forms processes. Double-click on any VBS file to view the script within a VB

script editor. See the Using Remote Scripting section for more information on remote scripts.

Note: The FormWare Internet Server Service must be running before any remote users attempt to log in to the File Server.

Follow these steps to configure the IFS settings, which are stored in the \FormWare\INI\System\ICapture.ini file:

1. Open FormWare Administration.

2. Select **Tools > Configure > IFS.** The FormWare Internet Server Configuration dialog displays.

3. If you are using I-Forms, select the appropriate options in the **Transaction Data Removal** area:

   o Select the **Empty Transactions** option that identifies whether to keep or delete empty transactions from the remote site. If you want to remove them, specify the number of hours to wait before removal occurs.

   o Select the **Active Transactions** option that identifies whether to keep or delete active transactions from the remote site. If you want to remove them, specify the number of hours to wait before removal occurs

   o Select the **Complete Transaction** option that identifies whether to keep or delete completed transactions. If you want to delete them, specify that they be removed when delivered to the JobFlow or after a certain number of hours.

4. In the **Web Server Connection Timeout** area, select the amount of time the remote client can try to connect to the Web Server before it fails. Select one of the following options:

   o Select **Never** if you do not want the connection to timeout.

   o Select **Timeout After** and specify the timeout period in milliseconds.

5. In the **Web Server Process Timeout** area, select the amount of time allowed before the connection times out due to a failed or hung process. Select one of the following options:

   o Select **Never** if you do not want the connection to timeout.

   o Select **Timeout After** and specify the timeout period in milliseconds.

6. In the **Transaction Directory** area, select the option that indicates where to store transaction data files before the transaction enters a JobFlow. Data files have either an IFX (completed transaction) or IFD (open transaction) extension. Select one of the following options:

   o **None** – No other location indicated. Data files are placed in the \FormWare\IFS\Datadirectory by default.

   o **Date** – Places data files in a subdirectory, named *<transaction date>* (based on the current date, mm_dd_yy format), in the \FormWare\Remote\Data directory. For example, if a transaction occurs on 3-22-02, FormWare creates a subdirectory named \Data\03_22_02 that contains all associated data files.

   o **Job** – Places data files in a subdirectory, named *<job name>* for the active job, in \FormWare\Remote\Data. For example, if a transaction occurs in the FWCLAIM job, the system creates a FWCLAIM subdirectory with all associated data files.

   o **Count** – Places data files in a subdirectory, named *<count>*, in \FormWare\Remote\Data\. Set the count increment in the field to the right—e.g., a 3 in the count field creates subdirectories in increments of 3 (3, 6, 9, etc.).

> **Note**: Copies of the data files delivered to JobFlow are also placed in subdirectories (with the same names as the database) under the \FormWare\Files directory.

7. The **File Relocation** settings determine where files go when remote users return them to the File Server. The FormWare Filler, Distributed Scan, Distributed XML, and Distributed WorkFlow components use these settings.

   o In the **BDF Location** field, type the path to the directory on the File Server where modified BDFs are to be placed when they are returned. The default location is C:\FWShare\FormWare\files\.

   o In the **Image Location** field, type the path to the directory on the host where modified image files are to be placed when they are returned. The default location is C:\FWShare\FormWare\images\.

8. In the **Remote Scripting** area, select the scripting language to be used for validating and authenticating remote users. This is required for remote users to gain access to the File Server via IFS. All scripts must reside in the \FormWare\IFS\Scripts directory. By default, a Sample.VBS is installed when you install the IFS. See the Using Remote Scripting section for more information. To identify the scripting language:

   o Select **Visual Basic Script** or **Java Script**. If you select Visual Basic Script, the Sample.vbs file, provided with the FormWare installation, is used as the default.

   o In the **Script Name** field, type or **Browse** for the name of the script you want to use.

   o Select the **Edit** button to create or modify a script using a script editor.

9. In the **FormWare Internet Server(s)** area, select the options that dictate how the Web Server connects to one or more IFS machines. To set these options:

   o Select the **Add** button to type a valid name in the Server Name list.

   o Press **ESC** or click outside the edit window to accept the name. You can also double-click on an existing server name to edit it.

   o Select **Delete** to remove the currently selected server name.

10. In the **Properties** area:

   o Select **Online** to indicate that the server is currently online. (If **Online** is not checked, the system considers the server to be offline and unable to process transactions.)

   o Select **Enable Logging** to force the server to generate log files for processed transactions. Log files contain historical data about the transaction, including errors. Log files reside in \FormWare\IFS\Logs and are named according to the workstation with an IFL extension (e.g., Server1.ifl). Double-click on the filename to view the log files.

> **Note**: If you use logging, periodically purge log files as part of system maintenance.

11. In the **Transaction Name Prefix** area, specify the prefix used to annotate the transaction data file names returned to the host. Valid values are:

   o **Machine Name** – Specifies that the prefix for the transaction data files corresponds to the server processing the transaction (e.g., Server1*nnnn*.ifd, where *nnnn* is a transaction ID unique to IFS Server1).

   o **Other** – Type the prefix for the transaction data file name in the field provided. If you are using only one IFS to process transactions, you can select **Other** and leave the prefix field blank. The prefix is primarily used to ensure the uniqueness of transactions across multiple servers.

12. Select **OK** to save your changes and return to Administration.

Using Remote Scripting

FormWare uses a remote script to validate and authenticate remote users. This script is required for remote users to gain access to the File Server via the Internet FormWare Server (IFS). All scripts must reside in the \FormWare\IFS\Scripts directory. You can write the script in VBScript or JavaScript.

The FormWare installation includes a sample VBScript, Sample.vbs, which validates remote users and processes remote operator statistics and data on the File Server. This default script uses the remote users' IDs and passwords and validates them against the user INI files on the File Server. You can modify the default script or create a new script to meet the authentication requirements of your specific environment.

Once you install IFS, the ICapture.ini file points to the Sample.vbs file by default. The IFS_SCRIPT_PATH parameter indicates the location and name of this script file.

See the Installing the IFS section for instructions on installing IFS.

Editing the Sample Remote Script File

To edit the Sample.vbs in the \FormWare\IFS\Scripts directory:

1. Open FormWare Administration.

2. Select **Tools > Configure > IFS.** The FormWare Internet Server Configuration dialog displays.

3. In the **Remote Scripting** area, type or **Browse** for the Sample.vbs file. When this name displays in the **Script Name** field, select **Edit**. The Sample.vbs displays in a text editor.

4. Refer to the TODO sections within this sample file and modify the file as necessary for the requirements of your environment. You can use the following methods in a FormWare Remote Script:

   o **GetProperty** - Retrieves a property value for the current transaction.

   o **SetProperty** - Sets a property value for the current transaction. Some property names are reserved (see below), while all others are user-defined and are stored along with the transaction data.

   o **LogMessage** - Sets a text string to display as a message window in the browser immediately before the next form displays. The text provided as an argument will be added to the Application Event log.

   o **CheckUserIni** - Validates the username and password, and generates a security token. The default code uses the user INI file.

   The following sample event handlers exist in the Sample.vbs:

   o **RmtScript_OnLogin()** - The system calls this event handler when any FormWare distributed product attempts to connect to the Web Server. The event handler obtains the user name by calling GetProperty("$USERNAME") and the password by calling GetProperty("$PASSWORD"). The event handler uses these values to determine if the requested login should be allowed. CheckUserIni, strUserName, strPassword, strSecurityToken validates the user name and password and generates the security token if these values are valid. If you want to allow the login, the event handler sets the security token to a non-empty value by calling SetProperty "$SECTOK", strSecurityToken. If the login should not be allowed, the event handler leaves the

security token unset. The event handler may optionally store information to indicate the login state.

- o **RmtScript_OnLogout()** - The system calls this event handler when any FormWare distributed product disconnects from Web Server. The event handler may do what ever is necessary to undo the optional login state actions that were taken in the RmtScript_OnLogin().

- o **RmtScript_OnValidate()** - The system calls this event handler when any FormWare distributed product attempts to access the Web Server after having previously successfully logged in, and prior to logging out. The event handler obtains the user name by calling GetProperty("$USERNAME"), the password by calling GetProperty("$PASSWORD"), and the security token by calling GetProperty("$SECTOK"). CheckUserIni, strUserName, strPassword, strSecurityToken validates the user name and password and generates a possibly new security token if these values are valid. The event handler uses these values to determine if the requesting user should continue to have access. If you want access to continue, the event handler sets the security token property to a non-empty (and possibly different) value by calling SetProperty "$SECTOK", strSecurityToken. If access should not continue, the event handler sets the security token to an empty string value.

- o **CheckUserIni(strUserName, strPassword, strSecurityToken)** - The system calls this event handler when a FormWare distributed product attempts to either initially log in or reaccess the Web Server after a successful login. The sample code within Sample.vbs uses the user INI file for security validation, therefore, the user INI must contain the $PASSWORD setting for the user in question. If it does, a default security token is generated. The sample code also assumes that the name of the user INI file and the $USERNAME parameter in that file are the same. This assumption is not made or required by any other part of the system.

- o **RmtScript_OnRecordOperatorStats()** -The system calls this event handler when Distributed Completion transmits operator statistics for a session from a remote client to the File Server. The sample code writes the operator statistics to a text file. See the Using the Operator Statistics File section for more information.

- o **RmtScript_OnRecordCustomData()** - The system calls this event handler when Distributed Completion transmits custom data files. If you have FPL or VB code on remote workstations that creates and updates custom data files with each processed batch, you can use this event handler to transmit this data to the File Server. The sample remote script retrieves this data based on the GetProperty("CLIENT_PATH"), GetProperty ("SERVER_PATH"), and generates a text file of the custom data.

5. **Save** the file, **Exit** the editor and return to the FormWare Internet Server Configuration dialog.

Note: The path to this script file is in the ICapture.ini file, under the IFS_SCRIPT_PATH= parameter.

6. Select **OK** to accept your changes and return to Administration.

Connecting a Remote Client to a File Server

FormWare Distributed Processing components connect remote clients to a File Server via the Internet. The first time a remote client connects to a File Server, the user is required to provide login information for a successful connection. After a successful connection has been established, a RmtXfer.ini file is created and stored on the remote client machine. This file contains connection information for the remote client so that the user does not have to login each time he connects to the server. If this file is deleted, the login dialog will be presented to the user again.

If you choose to, you can force remote operators to login each time they attempt to connect. See the Establishing Security for Distributed Processing section for more information.

Once you perform a remote workstation installation and install and configure all Distributed Processing server components on the host site, you can connect to the File Server from a remote workstation. This connection is accomplished using the Remote Sync Connection dialog. This dialog displays on the remote workstation for each Distributed Processing component used however, starting each component may accomplished in slightly a different manner.

The following steps explain how to perform a remote sync on a remote workstation used for more than one type of Distributed Processing task.

1. On the remote workstation, perform a remote workstation installation.

2. Run the Distributed Processing component you just installed. Based on your selection at installation the list of possible options are:

   o **Distributed Completion** - Includes Completion, VBA support, and the necessary remote transfer utilities.

   o **Distributed Scan** - Includes Scan Operator, Scan Designer, and the necessary remote transfer utilities. The system also selects the FWSLM required to run Scan Operator. Any remote client running Distributed Scan must be running the FormWare License Manager with the appropriate hardware key and license .DAT file installed.

   o **FormWare Filler** - Includes FormWare Filler and the necessary remote transfer utilities. You can optionally install VBA support by selecting the VBA option under FormWare Filler.

   o **Remote Transfer Utilities** - Includes the remote transfer utilities only. This option is primarily for transferring XML files, but you can also use it to transfer Batch Definition files (BDFs) in an ad-hoc fashion.

   o **FormWare Security License Manager (FWSLM)** - Includes the License Manager and License Monitor programs. This option is only required for use with Distributed Scan (Scan Operator).

3. At the Login dialog, type the user name and password of the remote operator. The Remote Sync Connection dialog displays.

4. In the **Web Server Name** field, type the name of your Web Server. If the remote client cannot resolve the computer name to the IP Address of the Web Server, type the IP Address of the Web Server.

Note: Your Internet administrator may have configured a firewall on the Web Server to prevent HTTP communication on port 80 and instead use a different port for HTTP communication. If you were directed to use a different port number, you must include that port number after the computer name or IP Address of your Web Server. For example, MyWebServer:4001 refers to port 4001 for HTTP communication.

---

5. In the **User Name** field, type the remote user's ID. This ID is used to locate the correct user INI file on the FormWare Server.

6. In the **Password** field, type the remote user's password.

7. If you are connecting to a central location, select the **Use Proxy Server** check box. Checking this option displays the **Proxy Server Address** and **Port** fields on this dialog.

   Connecting through a proxy server allows security through a secure socket. The proxy server filters and passes information from the remote client to the Web Server. Data transfer is seamless, and all functionality remains the same except that the user must know the name and/or IP address of the proxy server and the port number. Although this information is recorded in each user's INI file, it does not automatically populate the Remote Sync Connection dialog.

Note: Use a text editor to add the following parameters to the user INI file for each remote client user who needs to connect through a remote proxy server: RMT_USEPROXY=Y (or N to shut off capability), RMT_PROXYSERVER=<name of proxy server>, and RMT_PROXYPORT=<port number>. These parameters are embedded in the RmtXfer.ini file and downloaded to the remote workstation during the remote sync process. They are used for authentication and connection.

---

8. In the **Proxy Server Address** field, type the proxy server name or IP address.

9. In the **Port** field, type the port number for the proxy server.

10. Select **OK** to connect.

Configuring the Software

Verifying System Environment Variables

During installation of FormWare on the file server the setup program configures environment variables as appropriate to your system configuration without which FormWare processing will fail. For File System variables are configured in drive-path format (see table). On the File Server, the environment variables match the drive path to FormWare. On the client workstation, the variables reflect the UNC format. Following the installation of FormWare the system, the administrator for your FormWare environment may wish to modify the variables to reflect changes he has made.

The following environment variables are required at the machine system level for FormWare processing to be successful:

Variable Name	Variable Value (i.e., should point to)
FORMWARE	FormWare root c:\sharename\formware
CAERE InScript clients	c:\sharename\formware\caere
LITSYSDIR Mitek clients	c:\sharename\formware\mitek\sys
QSDIR Mitek clients	c:\sharename\formware\mitek

Variable Name	Variable Value (i.e., should point to)
PATH	Statement should always include: c:\sharename\formware\programs;\\computername\sharename\formware\bin
For InScript2 Recognition	PATH statement should also include: c:\sharename\formware\inscript2
For Parascript clients	PATH statement should also include: c:\sharename\formware\parascpt
For Full Text Recognition	PATH statement should also include: c:\sharename\formware\textbrdg

If these environment variables were not set during installation, you can set them by doing the following:

1. Right-click on the **My Computer** icon on the desktop.

2. Select **Properties**.

3. On the Advanced tab, select the **Environment Variables** button.

4. In the System variables area, select the **New** button.

5. Type the **Variable Name** and **Variable Value** in the fields provided and select **OK** to set the value.

6. Select **OK** two more times to save the values and exit.

Running the Administration Module

The Administration module is the entry point into most FormWare design and maintenance processes. To open the Administration module from your desktop, select **Start > Programs > FormWare > Administration**.

Starting the License Monitor

When you install FormWare on a network, the software is installed on the File Server and on one or more client workstations. For each network installation, you must have a blue hardware key and a diskette containing a license file (*.dat). The license file contains information about the number of licenses for each FormWare component you purchased and governs network user access. You must attach the blue hardware key to the back of the File Server. You must also copy the license file from the diskette to the \FormWare\Other\SLM directory.

To view the active components within the FormWare system (on the network), do the following:

1. Open the Administration module.

2. Select **Tools > License Monitor**. The FormWare License Monitor dialog displays.

    The Active Clients tab displays information relative to each active FormWare component and the workstation or server on the network where it is running. It contains the following information about the clients running on the network:

    o **Client -** Displays the name of each FormWare component currently running on the network.

    o **Machine -** Displays the name of each workstation (either remote or server) where a component is running.

- **Login Time -** Displays the date and time that each component was activated (logged on to the network) using the License Monitor program.

At the bottom of the License Monitor window, you see the following fields:

- **Location -** Displays the location of FormWare directory files (typically on the workstation where FormWare has been installed). The License Monitor program (FWSLM.exe) may or may not be running on that particular workstation or server.
- **Started -** Displays the date and time the License Manager was started.
- **Hardware Key -** Displays the unique code programmed into the hardware key (a hexadecimal number). The hardware key resides on the workstation or server where the FormWare system was installed and is running the License Monitor. The code in this field should match the hexadecimal codes in the license DAT file.
- **CPU Utilization -** Displays the percentage of CPU usage for the License Monitor (i.e., how many components are logging into and out of the License Monitor).

Note: The CPU information in the license monitor applies only to Windows NT. Windows 98 does not enable utilization information by process. For Windows 98, this field indicates "N/A'" (i.e., not applicable).

3. To see information relative to the FormWare license for each type of component on the network, select the System Data tab. This dialog displays the following information:
    - **Client -** Displays the name of the FormWare component.
    - **Active -** Displays the number of workstations actively running the licensed component.
    - **Allowed -** Displays the number of locations allowed to run the component based on your license.
    - **Logins -** Displays the number of successful logins to this component based on your license.
    - **Failed -** Displays the number of unsuccessful logins to this component based on your license. Typically, a value in this column indicates that the maximum number of allowed logins was exceeded.
    - **Expires -** Displays the license expiration date, if any, for the component.
4. To start the License Monitor, select the **Start** button. This button is only active if the program is idle.
5. To stop the License Monitor, select the **Stop** button. Use **Stop** with caution. When you select **Stop**, you also log off all network users of FormWare components. Although the components continue with any current processing, users will be unable to access the components upon subsequent attempts to log in.
6. To refresh the list of components and related data, select the **Refresh** button.
7. When you have finished monitoring the licenses, select **Close**.

Creating Import and Export Devices

To create an import/export device:

1. Open FormWare Administration.
2. Select the Devices directory from the File Browser.

3. Right-click in the Contents area and select **New** from the context menu. The Device Properties dialog displays.

4. In the **Device Name** field, type the name of a device.

5. In the **Path Name** field, type the full path and name (with extension) of the file that is created when this device is specified. To specify a printer, use LPT1 or the network printer name. You cannot specify a device name that has the name LPT1 and an extension as a file. LPT1, LPT2, LPT3 and LPT4 are reserved and are assumed to be print devices. You can use the %FORMWARE% environment variable to simplify the path name.

6. If you are using JobFlow to export batches, type the name of the batch log file with a .BLF extension in the **List Name** field. This causes the Administration program to generate a batch log file with the name of every batch to export. You can use the %FORMWARE% environment variable to simplify the path name.

7. In the **Process** field, specify the program to be executed by FormWare once data are written to the device. This could be a communications package or other gateway software to upload the data to a mainframe. This option applies only to export devices–not TYPE devices–in an export program.

8. Set the appropriate options for the device. Device options control how data are written to the device. Valid options are:

    o **Append to Device -** Data exported to this device are appended to the device if the file exists.

    o **Mark as Output -** The BDF is modified and marked as output. The BDF contains binary data—information captured from a batch of paper forms, either by recognition technology, manual data entry, or a combination of the two.

    o **Out Extension –** Writes output statements in the program to a file that has the batch name with an .OUT extension. This is for export programs only.

    o **Print Device -** Appends a <CR><LF> to the end of each line when exporting. Removes a <CR><LF> from the end of each line when importing.

    o **Numeric Extension -** Enables you to create a file with a numeric extension. Each time data are exported to this device, the number is incremented by one. In the **Path Name** field, type the name of the file with a 000 extension.

9. If you are using the JobFlow software, set the appropriate List Options for the device. The values in this section relate to the name you typed in the **List Name** field. Valid list options are:

    o **Count -** Indicates the number of batches to group for export. The combination of the Count and Time field values allows the JobFlow software to group batches for export and purge redundant or superfluous files.

    o **Time -** Indicates the time for exporting. The time value must be in HHMM military time format (1300 is 1:00 pm).

10. Select **OK** to accept your changes and return to the Administration window.

Setting Client Display Properties

You can change the desktop display settings on computers that have a Microsoft Windows operating system. Display settings control how the desktop looks and displays information. Display settings also have an impact on how forms are displayed within a FormWare application.

If you are designing forms using Designer, the display settings you use on the computer where the form is created must be identical to the display settings that Completion operators use on their workstations. If you do not use the same display settings at design-time that are used at runtime, you may get unexpected or inaccurate results. For example, if you design a form on one computer that has display settings of 1024 by 768 pixels and Completion workstations use display settings of 800x600, the forms will not display correctly and the operators cannot view the Completion window as it was designed.

If you cannot design forms using the same display settings as the runtime computers, then use the lowest display settings possible on the design computer. This ensures that the runtime computers (using Completion) will display the entire form without inadvertently cropping the image in the viewport or the data entry display.

To change desktop display settings on a computer with a Windows operating system:

1. From your desktop, right-click to display the context menu and select **Properties** to display the Display Properties dialog.

2. Select the Settings tab on this dialog.

3. In the **Screen Area** section, move the scroll bar to the desired setting.

Note: This is called the Desktop Area section in Windows NT.

4. Select **Apply** to apply the new desktop settings. A dialog displays that informs you of the desktop display test process, which takes a few seconds. Under Windows NT, the desktop display testing process is slightly different. Follow the appropriate prompts if you are on an NT machine.

5. Follow the prompts to accept or reject the new settings.

6. Select **OK** to accept your changes and exit the dialog.

Using Desktop Shortcuts to Launch File Server Components

To launch FormWare components from a workstation, you can create shortcuts on the desktop. You can also launch FormWare components that exist on a File Server from a remote workstation using desktop shortcuts. In this situation, the shortcut pointer typically contains a drive letter that has been mapped to the File Server.

Be careful not to change the drive letter or File Server location or your remote shortcuts will not be valid. For example, assume your File Server, FWServer, has a shared directory called FWShare. A remote workstation points to this server location (\\FWServer\FWShare) using "F" as the mapped drive letter. The remote workstation also has a shortcut to this mapped drive and the appropriate FormWare component (e.g., F:\FormWare\Programs\<executable name>). If you

move your FormWare application to a new server called NewServer, you would then have to map the F drive on the remote workstation to the new server (i.e., \\NewServer\FWShare).

Caution: Microsoft provides "smart" shortcuts, which ignore your drive map changes. In the previous example, smart shortcuts would assume that because your original shortcut was created with a drive letter mapped to FWServer, you still want the shortcut to point to that server. It would disregard your drive mapping change to a new server and run the programs from the old server. Therefore, if you change the drive mapping to point to a new server, you must delete your desktop shortcuts and re-create them to point to the new server.

---

3 Designing

Designing Overview

Welcome to the FormWare Designer module. The software functions and tools within this module enable you to create, edit, and test FormWare applications. Within this section, tasks are ordered in the logical sequence you would perform to create a basic FormWare application.

Refer to the Designing Checklist section for a high-level list of the procedures necessary to design your FormWare application.

Designing Checklist

The purpose of the Designing Checklist is to familiarize you with some of the more common tools you will use to create a FormWare application. The checklist items follow the typical creation steps for a simple job. For further clarification of any procedure, refer to the appropriate section in this FormWare Guide.

To use this checklist for a sample job, use a form with few data capture fields and one that has no extraneous marks, such as smudges, speckles or tears. In addition, these steps assume you have a scanner attached to your design station.

Starting FormWare

- On the desktop, select **Start > Programs > FormWare > Administration**. The Administration window displays.
- Select the **Designer** button to open Designer.

Creating a New Job

- Open Designer.
- Select the **File > New > Job**. Type DEMO for the new job name.
- Set Job Defaults. You can leave the selected defaults. Note that once the Form parameters are set, they cannot be changed for this job. You can change other settings later in either Form or Field Properties.
- Select the **Push Pin** button to keep the Job Properties dialog in focus.
- Select **Edit > System Defaults > General**. Deselect **View Tests in Completion.** By deselecting this option, you force test results to display either in fields on the Designer window data form or within a component progress dialog.
- Select **File > Save All** to save your work.

Setting Scanner Properties

- In Designer, select **Job Properties > Scan > Properties** button to display Scan Designer.
- In the **Job Name** field, select DEMO from the drop-down list.
- Use the default settings.
- Select **File > Save As** and type DEMOSCAN for the scan configuration file name.
- Close Scan Designer.
- Save your work.

Importing Images

- Select approximately 50 of a single form type to process.
- Open Scan Operator.
- Select **Config** and select DEMOSCAN.scg from the list.
- Select **Source** to set up source of scanned images.
- Select **Start** to scan images into BDFs.
- Review each listed image to find the straightest and clearest image to use as the setup image and note the name.
- Close Scan Operator and return to Designer.

Selecting a Setup Image

- Select **Form Properties > Image > Browse** next to the **Setup Image** field to locate and select the image to use as the setup image.
- Set the magnification of the image in the Image Viewport using either:
  - The **Horizontal Scale, Vertical Scale**, and **Rotation** fields to manually adjust the size of the image in the Viewport.

  Or,
  - The **Fit Window** button, which automatically resizes the image in the Viewport to fit completely within the Viewport window.
- Select **Apply** to accept your changes and view your results on the Designer window.
- Save your work.

Applying Image Pre-Processing

- Select **Form Properties > IPP > Properties > Image Deskew > OK**. This will straighten your image.
- Select Form Properties > Image > IPP. Test the IPP setting on your setup image. If you are satisfied with the image in the viewer, select **OK** to use this setup image.
- Save your work.

Defining Form ID

- Right-click on the image and adjust the display of the setup image so the feature you want to use as a Form ID feature displays.
- Select **Form Properties > ID > Type = Field** to determine the type of Form ID.
- On the Field Type toolbar select the **Form ID** button and then click on the form. A one-character, green field with handles displays on the data form.
- Click on the image and draw a box around the Form ID anchor feature. This is the Form ID zone.

- Select **Field Properties > ID > Features** to display the Registration Anchor dialog with the ID feature zoned in the Image Viewer.

- Select the **Test** button on the toolbar to run ID processes on the anchor. If the test is successful, the anchor point displays a small box outlined in red.

- Select **OK** to accept your new Form ID definition and return to the Field Properties dialog.

- Save your work.

Creating Registration Anchors

- Right-click on the image and adjust the display of the setup image so the feature you want to use as a registration anchor displays.

- On the Field Type toolbar, select the **Registration** button and click on the data form grid to place a Registration field. A one-character, blue field with handles displays on the data form.

- Click on the image and draw a box around the Registration anchor feature. This is the registration zone.

- Select **Field Properties > Registration > Registration** to display the Registration Anchor dialog with the registration featured zoned in the image viewer.

- Select the **Test** button on the toolbar to run registration processes on the anchor. If the test is successful, the anchor point displays a small box outlined in red.

- Select **OK** to accept your new anchor definition and return to the Field Properties dialog.

- Create at least one text anchor or three of any other types of anchor for each for form.

- Save your work.

Creating Fields

- Right-click on the data form area of the Designer window and select **New Field > Any Type** from the context menu. The cursor changes to a cross-hair. Drag the mouse diagonally across the data form and release the mouse to place the field. To reposition the field, click on it once and drag it to the correct location.

- Double click on the field and select the **Field Properties > General** tab. All properties dialogs in Designer can be forced to display using the push-pin button in the upper left corner of the dialog.

- Highlight the default name in the **Name** field and type a new name for this field.

- Resize the field on the data form by clicking on the field handles and dragging.

- With the field selected, click on the image in the Image Viewport and draw a box around the data you want to capture in that field.

- Select **Field Properties > IPP > Auto Remove Background** to automatically select removal features. Select **Test** to view the results.

- If there are still some features remaining, select **Field Properties > IPP > Enhancement** to manually set removal properties.

- Select **Field Properties > Recognition** to set the recognition values for the field.

- Create a label for this field. Right-click on the data form and select **New Field > Label**. Type the label description in the **Text** field on the Field Properties, General tab.

- Save your work.

- Repeat for all the data capture fields on your image.

Defining Enhancement

- Select **Field Properties > IPP > Enhancement** to display the Removals Features dialog
- On the toolbar, use the tools to select the type of feature you want to remove.
- Select the **Properties** button to further define the removal.
- Repeat for each feature.
- Select the **Test** button. Now only the data you want to capture displays.
- Save your work.

Defining Recognition

- Select **Field Properties > Recognition > Errors Allowed** = 3.
- Select **Field Properties > Recognition > Recognition Type** = Machine Print or Hand Print (format of the data you are capturing).
- Save your work.

Testing

Testing a single process is referred to as *discrete testing*, while testing several processes simultaneously is referred to as *comprehensive testing*. Discrete testing is applied to a single image, while comprehensive testing can be applied to a batch of images. Comprehensive testing is conducted on all levels within the Designer module: job, form, and field. Discrete testing is conducted only on the form and field levels.

- To perform discrete testing, select the property page for the process you want to test. For example, if you want to test only recognition on the form level, then access the Form Properties dialog, Recognition tab and select the **Test** button.
- To perform comprehensive testing, select Test tab on any of the Job, Form, or Field Properties dialogs. You can select multiple processes to execute in the test and you can specify a batch of images for the test.
- Save your work.
- If any of your test results are not as you want them, you can go back and fine-tune any of the settings.

After creating, testing, and fine-tuning your job, you are ready to create a JobFlow and then start processing.

Defining Scan Settings

The FormWare Scan module provides advanced scanning capabilities for successful batch-based forms processing applications. The Scan module is designed for use with all of the components of the FormWare software, creating Group4 TIFF images and Batch Definition files (BDFs) needed for a seamless forms processing system.

When using FormWare Scan with ISIS, the ISIS interface enables you to access the third-party scanner settings, but you should refer to your scanner software for configuration details. Settings for specific scanner features vary by scanner.

Understanding Scanning

FormWare supports two scanning methods:

- **Local scanning** enables you to scan documents on scanners and import images from local files. Use local scanning processes when your server and your scanner hardware reside in one physical location.

- **Distributed scanning** enables you to scan documents on remote scanners and import images from remote files. Use distributed scanning processes when your server and your scanner hardware reside in separate physical locations. See the Defining Distributed Scan section for more information.

There are two components that control the local scanning processes:

- **Scan Designer** - Lets you create or modify any number of different scan profiles, which are saved as Scan Configuration (SCG) files. Scan profiles contain all of the settings needed to perform a scanning job.

- **Scan Operator** - Lets you select SCG files, scan documents, create BDFs, import image files, and add batches to a JobFlow. Using the Scan Operator module, you can scan paper documents, import faxes and copy existing images from one folder to another.

During the scanning process, the image created from each scanned document displays on the screen. The scan operator can identify problems such as failed scanner bulbs or improper document orientation and stop the scanning process to fix the problem. Once the scanning process resumes the new images overwrite any existing poor-quality images.

You can also use the Scan module to scan hundreds of paper documents for batches and create the BDFs used in Image Pre-Processing, Recognition, and Completion. The BDF contains a list of the image file names within the batch. You designate how the scanned forms are divided to create BDFs (e.g., scanning a specified number of forms, encountering a separating blank piece of paper, or identifying a feature such as a bar code).

In addition to creating BDFs, the Scan module can send batches to the JobFlow system that manages the flow of forms processing. When you add batches to a JobFlow, you can assign them a registration or priority number. Using registration numbers, you can group batches together for monitoring or export. Using priority numbers, you can order batches so the most important batches are the first processed regardless of the scanning order. Once the Scan module adds a batch to the JobFlow system, it is immediately routed to the first process in the JobFlow.

Using Supported Drivers and Scanners

Because ISIS is a de facto standard and the basis of an industry standard (ANSI/AIIM MS61-1996), ISIS users are assured that their imaging applications will drive a wide variety of scanners. The list of supported scanners changes continually. Obtain an up-to-date list of supported ISIS scanner drivers at www.scannerdrivers.com.

Running ISIS over Kofax

FormWare v4.6 and earlier versions were developed based on the Kofax ImageControls toolkit. As a result, FormWare v4.6 and earlier versions required users to purchase a Kofax card to communicate with their scanners. FormWare v4.7 and later have been re-engineered to use the PixTools (i.e., ISIS) Toolkits created by Pixel Translations. As a result of this re-engineering, FormWare now supports more scanners and more interfaces, including USB, 1394, and Adaptec SCSI. With this new version of FormWare, a Kofax card is no longer required for SCSI scanners.

For users who wish to take advantage of their existing Kofax Adrenaline cards, the 5.0 version of FormWare supports an ISIS driver that communicates with Kofax Adrenaline 850V and 1700V cards.

Using Kofax Cards

Most ISIS drivers communicate with the scanner over an advanced SCSI programming interface (ASPI). The supported SCSI cards are mostly Adaptec cards. No ISIS driver communicates directly with Kofax Adrenaline 1700V and 850V cards, but there is an ISIS driver that communicates with ImageControls that allows a user to continue to use 850V/1700V cards. The other Kofax cards, the Adrenaline 450, 650, and 650i etc., are just SCSI cards. ISIS drivers that communicate through ASPI can be used with these cards with no additional software.

Note: Pixel Translations develops ISIS drivers that can communicate with Kofax video cards. For a complete list, see www.scannerdrivers.com.

Using Special ISIS Drivers

The following ISIS drivers communicate with the scanner through another toolkit:

- kofaxic3.pxn - Communicates with ImageControls and will work only with the Kofax 850V and 1700V Adrenaline cards. ImageControls handles the communication to the scanner.
- pixtwain.pxn - Communicates with any TWAIN driver. The TWAIN driver handles the communication to the scanner.
- kofaxbic.pxn - Communicates with the Kofax KIPP API and works with the older Kofax cards. However, these cards have been discontinued. The KIPP API handles the communication to the scanner.

Current FormWare customers who want to use the new FormWare Scan with their Adrenaline 850V or 1700V cards need the kofaxic3.pxn driver so that they can continue to use their Adrenaline card and their scanner.

Working with Images

After the paper documents are ready, the next step is scanning the documents to produce electronic images. You should scan the documents using a method that produces the cleanest possible image. For example, images scanned at 400 dots-per-inch (DPI) produce better recognition results than images scanned at 200 DPI. However, 400 DPI images require significantly more disk space and recognition processing time. For optimal results, set the scanner to the lowest DPI that creates a readable image.

Another important part of the scanning process is adjusting the contrast and intensity to improve the quality of images, reduce specks on images, and remove light, unwanted lines. You can also scan a portion of the document instead of the entire document, resulting in a smaller, less-cluttered image.

Examples: On the first image below, the scanner is not adjusted correctly, which causes specks or "noise" to appear on the image. On the second image, the scanner contrast and intensity have been adjusted to produce a cleaner, more readable image.

How the paper document is inserted into the scanner also impacts the quality of the image. For example, perforations can cause the scanner to grab the document unevenly and can skew the image. Therefore, you should not scan documents with a perforated edge, with that edge inserted first into the scanner feeder. If a paper document has a perforation, rotate the document and feed it into the scanner with the clean edge first. Later, you can rotate the image to the proper orientation on the screen.

Although you can rotate the image during the scanning process, remember that each additional step in the scanning process decreases scanner performance (the number of paper documents scanned per minute). Therefore, we recommend that after scanning, you rotate images during Image Pre-Processing.

Starting Scan Designer

There are several ways to access the Scan Designer module, depending on which one you want to run.

FormWare Scan with Kofax

- From the Tools menu in FormWare Administration, select the **Scan Designer** option.
- From the toolbar in FormWare Administration, select the **Scan Designer** button on the toolbar.
- From the Start menu, select **Start > Programs > FormWare > Scan Designer**.
- From your desktop, select the Scan Designer icon (if you have set up this program as a shortcut on your desktop).

FormWare Scan with ISIS

- From the \FWShare\Formware\ISIS, select the Scan Designer program.
- From your desktop, select the Scan Designer icon (available only if you have set up this program as a shortcut on your desktop during or after installation of FormWare Scan with ISIS).

Selecting a Scanner

Before you can scan, you must select your scanner. To select a scanner:

1. Turn on your scanner.

2. Launch QuickScan.

3.   From the Scan menu, select the **Select Scanner** option. QuickScan takes a few moments to determine the names of the scanners supported by all of the installed scanner drivers and then displays the Scanner Selection dialog.

Note: If you are using a scanner that uses a TWAIN driver, you must first install the driver (provided by the scanner manufacturer). If the TWAIN driver does not appear in the Scanner Selection dialog, install the scanner driver and scanner. After you have installed the TWAIN driver, choose the scanner name that includes **w/PixTWAIN** from the Scanner Selection dialog (or select **Any Scanner w/PixTWAIN**) and select **OK**. QuickScan will run the PixTWAIN Configurator, which verifies that the TWAIN driver can work with QuickScan. The PixTWAIN driver allows scanners that use TWAIN drivers to work with ISIS applications such as QuickScan.

4.   Select your scanner from the list displayed in the Scanner Selection dialog.

Note: If you are using Kofax video interface cards, be sure to choose a scanner that includes Kofax Kipp for older video cards (e.g., 9250, 9275, 7300) or w/Adrenaline for Kofax Adrenaline video interface cards. If you are using a Kofax Adrenaline video interface card or the older KIPP video interface card, you must first install Kofax's hardware runtimes (ImageControls for the Adrenaline and KIPP Toolkit for the KIPP boards).

5.   Select **OK** to close the Scanner Selection dialog box.

To create a scan configuration file, you define the following types of properties:

- JobFlow Properties
- Scanner Properties
- File Properties
- Import Properties
- Data Properties

Setting JobFlow Properties for Scanning

By setting JobFlow properties for scanning, you enable operators to add a new BDF to a JobFlow task during an importing or scanning operation. By setting options on this window, you create unique Scan Configuration files (SCG files). These files are stored in the \FormWare\Scan folder. To set the JobFlow properties for scanning, do the following:

1.   From FormWare Administration, select the **Scan Designer** button from the toolbar. The Scanner Configuration Designer window displays.
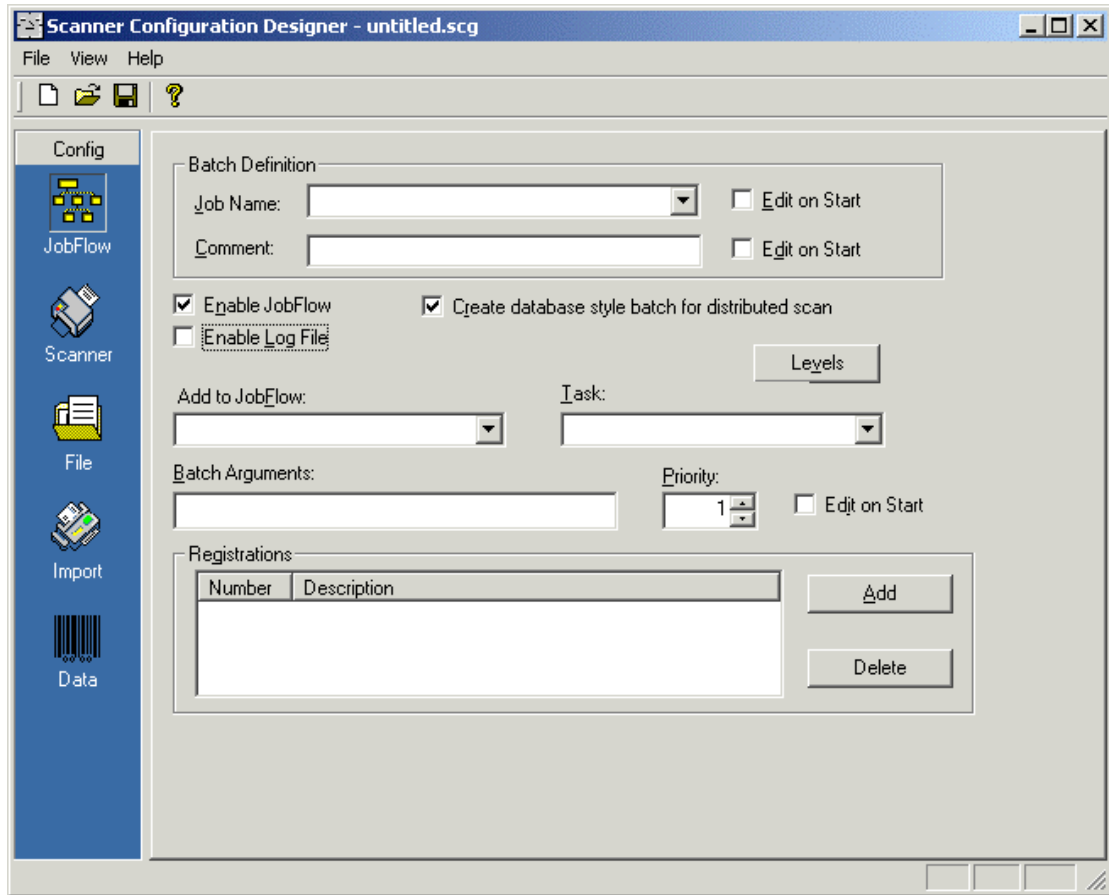
2. In the **Job Name** field, select the name of the Job or type a new job name for the job associated with the images in the BDF. All properties set for the job are applied to the images.

3. Select the **Edit on Start** check box if you want the operator to be able to change the job when starting Scan Operator.

4. In the **Comment** field, type any remarks you want to write into the BDF. Type a semi-colon between comments to place each comment on a separate line.

5. Select the **Edit on Start** check box if you want the operator to be able to change the comments when Scan Operator starts.

6. Select the **Enable JobFlow** check box to assign BDFs to a JobFlow and JobFlow task. Selecting this option enables the remaining fields on this dialog. Typically you return to this window to check this value after you have created your JobFlow.

7. Select the **Enable Log File** check box to force the Scan Operator to create a log file that lets Captiva Support evaluate the design of your job. The log file is saved as <ConfigFileName>.log in the \Windows\Temp directory. Log files grow large very quickly, so you typically create log files only during testing and not in a production environment.

Note: The Scan Operator log file is for Captiva diagnostic and testing use only and may be eliminated in the future.

> **Note**: *Do not* select the **Create database style batch for distributed scan** check box if you are running FormWare File System. This check box is for use *only* when you are running a Distributed Scan installation with a FormWare Database system.

8. In the **Add to JobFlow** field, select the JobFlow Definition file (JFD) to which the new BDFs are assigned if you selected the **Enable JobFlow** check box.

9. In the **Task** field, select the JobFlow task to which the BDFs will be assigned if you selected the **Enable JobFlow** check box. The JobFlow tasks are listed in order of execution within a JobFlow.

10. In the **Batch Arguments** field, type any client parameters that need to be passed to the JobFlow Client. See the Client Parameters section for a list of available parameters.

11. In the **Priority** field, select the numeric value that determines which BDFs are processed first for the JobFlow task. The default value is 1, the lowest priority, while the highest priority that can be assigned is 999.

12. Select the **Edit on Start** check box if you want to allow the operator to be able to change the batch priority when starting Scan Operator.

13. In the Registrations area, select the **Add** button to add registration numbers to group and monitor BDFs during the JobFlow process. In the Registration Numbers dialog that displays, highlight a number and select **OK** to assign that registration number to the new BDFs.

14. Select **Save** on the toolbar to display the Save As dialog and save the new scanner configuration with a unique name.

15. Select **Exit** from the File menu to exit the Scan Designer module.

Setting Scanner Properties

You can use the Scan module to scan a single paper form, which allows you to test scanner parameters such as contrast, intensity, and dots per inch. The Scan module supports single or duplex tray scanning, portrait or landscape orientation, and most common paper sizes. Support for these options, as well as for contrast and intensity, depends on the type of scanner and device card you use.

FormWare does not check the options available on your scanner hardware. Therefore, check the documentation that came with your scanner to determine which options are available to you. If your hardware does not support an option that is available on the Scanner Properties tab, then do not select that option.

The following tabs are available on the Scanner window and enable you to set options that apply to your scanner:

- Scanner Settings - scanner hardware, including page orientation and paper size.
- Image Processing Settings - rotating images and selecting a portion of an image to save.
- Endorsement and Annotation Settings - endorsing the documents and images and configuring the image font.

Viewing Scanner Properties (ImageControls)

If you have the Kofax ImageControls installed on your machine, you can use its VCDemo program to view the properties available on your scanner.

Note: To install the Kofax ImageControls on your computer, select **Install Scan Drivers > Install KF-Series Runtime (v 3.2)** option on the FormWare installation CD.

Perform the following steps to use the VCDemo program to see your scanner's properties:

1. Select **Start > Programs > Kofax ImageControls > VBDEM32P.EXE** option on the Windows Start menu. The VCDemo (32-bit) for ImageControls dialog displays.

2. Select **Source > Scanner** from the menu bar to display the Select Scan Source dialog.

3. Select your scanner from the **Source** field and then select **OK** to return to the VCDemo (32-bit) for ImageControls dialog.

4. You can also set scanner properties for your scanner using options on the Source menu:

    •   Select **Source > Properties** to set the initial properties for your scanner.

    •   Select **Source > Advanced Properties** to set advanced properties for your scanner.

    •   Select **Source > Options** to set additional properties for your scanner.

5. Select **Save** on the toolbar to save your changes.

Defining Scanner Settings

To set the scanner settings for your scanner within the FormWare Scan module:

1. From FormWare Administration, select the **Scan Designer** button. The Scanner Configuration Designer window displays.

2. Select the Scanner icon on the Config panel. The Scanner Settings tab displays.

---
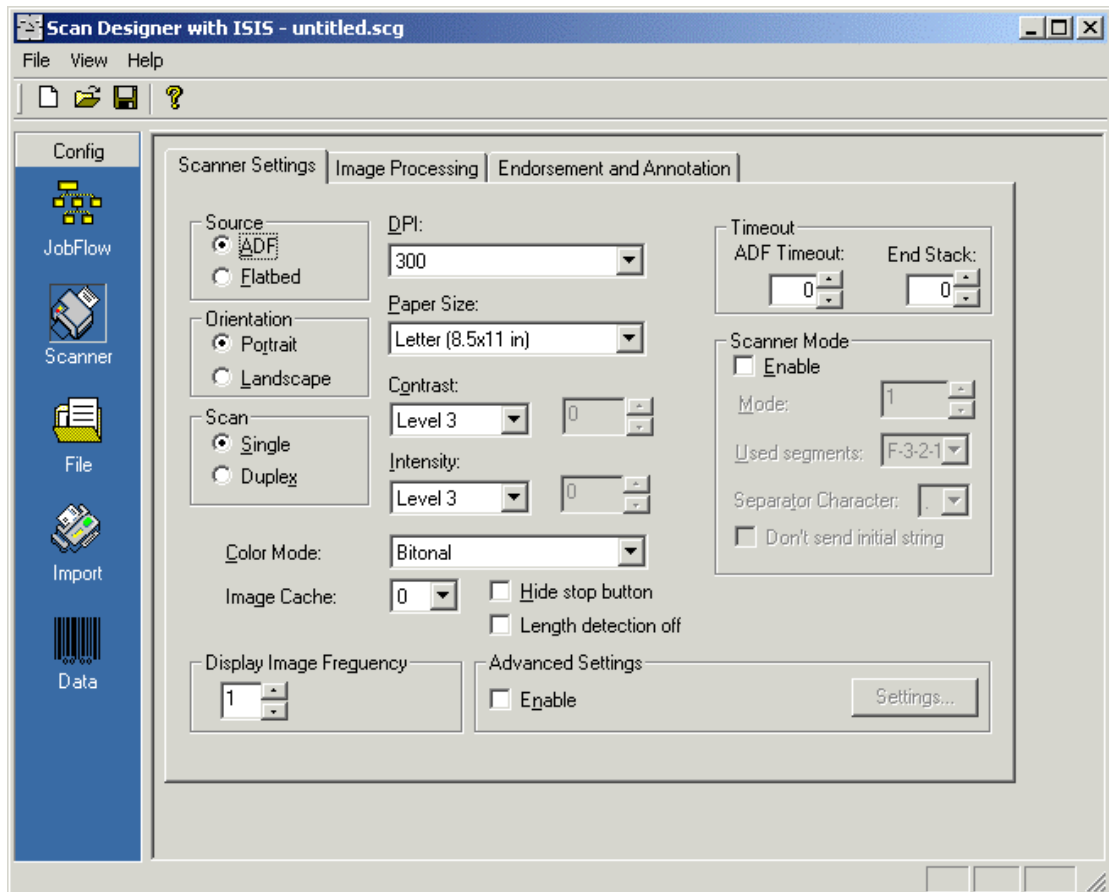
3. In the Source area, select the option that determines how documents are fed into the scanner. Most high-performance scanners use an Automatic Document Feeder (ADF). Additionally, most scanners are either ADF or flatbed, while only a few types of scanners support both. Valid values are:

    - **ADF** - Your scanner scans a stack of documents, one after another, at high speed.

    - **Flatbed** - Your scanner has a glass top upon which you set a single document to be scanned. After each document is scanned, a message displays informing the operator to place a new document on the scanner.

4. In the Orientation area, select **Portrait** to scan in portrait orientation, which is the standard orientation for an 8½-inch x 11-inch document or **Landscape** to scan in landscape orientation (i.e., 11-inch by 8½-inch).

5. In the Scan area, select **Single** to scan only one side of a document and create a single image file for each document or **Duplex** to scan both sides of a document (if your scanner supports Duplex scanning), and create a separate image file for each side of the document.

6. In the **DPI** field, select the number of dots used within an inch of space on an image. A higher DPI setting indicates that the dots are smaller, resulting in a higher quality, more readable image. The DPI determines the size of the image stored on disk. Because disk space is a premium with high-volume forms processing, the DPI you select is extremely important. Higher DPI settings result in larger file sizes. For this reason, select the lowest DPI that still creates a clean, readable image. The default value, 300 DPI, can produce a readable image on most types of forms.

7. In the **Paper Size** field, select the dimensions of the document being scanned. Selecting the correct paper size ensures that all of the data on the document is included on the image, and that large, white spaces are not inadvertently inserted on the right and bottom portions of the image.

8. In the **Contrast** field, select the contrast that provides the best range between the lightest and darkest tones on your paper documents. Adjusting the contrast can improve an image's readability by removing speckles or combs, and therefore increase the recognition accuracy rate. The default contrast setting is Level 3. You can select from Level 1-8 or from the following:

    - Auto – Scan Designer automatically sets the contrast.

    - Panel – Set the contrast manually using the scanner hardware settings.

    - Percent – Set the contrast as a percent. When you select this option, the percent field next to the **Contrast** field becomes active. Set the contrast percent using the arrow keys or by typing a number.

9. In the **Intensity** field, specify the brightness of the bulb used during the scanning process. Adjusting the intensity can reduce noise on the scanned image. The default intensity setting is Level 3. You can select from Level 1-8 or from the following:

    - Auto – Scan Designer automatically sets the intensity.

    - Panel – Set the intensity manually using the scanner hardware settings.

    - Percent – Set the intensity as a percent. When you select this option, the percent field to the right of the **Intensity** field becomes active.

Note: Some scanners do not support the **Intensity** setting.

10. In the **Color Mode** field, select a color image storage option that determines how many colors or shades of gray your scanner uses when translating scanned pages into images. Select a resolution that your scanner supports. Valid values are:

   - Bitonal **-** Translates images into black and white at one bit per pixel.

   - Millions of Colors **-** Translates images into up to 16 million colors at 24 bits per pixel: eight red, eight green, and eight blue.

   - 256 Gray **-** Translates images into up to 256 shades of gray at eight bits per pixel.

   - 16 Gray **-** Translates images into up 16 shades of gray at four bits per pixel.

11. In the **Image Cache** field, select the number of images cached during the scanning process. An image cache temporarily holds image files. When you increase the cache amount, scanner performance increases. However, if you set to a value higher than the default of 0 and the operator cancels the scanning process, any images stored in the cache are lost and the documents must be rescanned.

12. Select the **Hide stop button** check box to hide the **Stop** button on the Scan Operator window when scanning on a high-end Kodak scanner. This requires the operator to stop the scanning process from the Kodak scanner rather than from Scan Operator.

13. Select the **Length detection off** check box to turn off the automatic length detection feature of your scanner. Some scanners have sensors that detect the actual length of the page scanned and produce an image of that length. If you turn off this feature, the scanner creates an image of the set paper size.

14. In the Timeout area, specify time limits for scanning images. Kodak scanners must have both the **ADF Timeout** and **End Stack** options set to zero. Please refer to your Kodak Manual for further configuration information. For ISIS scanners, you can set the following options:

   - **ADF Timeout -** Specify the number of seconds the scanner is allowed to process each image before proceeding on to the next hardcopy document. If the timeout limit occurs before a particular image is fully scanned the scanner ceases processing the current image and continues on to the next hardcopy document in line.

   - **End Stack** - Specify the number of seconds the scanner is allowed to process each image before the current scan session stops and a Time Out error occurs. If an image is not scanned in the set period of time, a "Time out error" occurs. When you set **End Stack** to zero, most scanners will ignore this setting and scanning will process without errors.

15. In the Scanner Mode area, you set the initial value of each transaction string as required by Kodak scanners. Set the following values:

Note: The Scanner Mode options are required when you are using a Kodak scanner with modes enabled. Use these options only with Kodak scanners.

   a. Select the **Enable** check box to activate the Scanner Mode options.

   b. In the **Mode** field, select the scanner mode number based on the configuration documentation provided with your scanner. You can set modes 1 to 18 on Kodak scanners, which correspond to specific imprint string configurations. For example, Mode 1 could be configured for a 3.5 string, while Mode 2 could be configured for a 4.4.2 string. This value should match the scanner mode number you selected using the Kodak mode setup software.

   c. In the **Used segments** field, select the used segments value based on the configuration documentation provided with your scanner. All options display, not just the options supported by your scanner.

    d.   In the **Separator** field, select the separator character you want to use between string segments.

Note: When using the Scanner Mode options, you must create imprint string segments that correspond to the Used segments and Separator configurations you select. For example, if you select F-3-2-1 in the **Used segments** field, you must create at least one imprint string segment for the fixed field designated by the "F," one imprint string segment each for the fields 3, 2, and 1, and one imprint string for each separator.

---

    e.   Select the **Don't send initial string** check box to set the imprint string at the scanner. The operator must then enter the imprint string on the LCD panel of the scanner. Scan Operator then allows the Image Address Seed to serve as the imprint string for the scanning process. The Image Address Seed is the scanner-derived string from a Kodak scanner, which overwrites the imprint string each time an image is scanned. In most cases, the imprint string is defined in Scan Designer and passed to the scanner. For a Kodak scanner, the string is generated by the scanner and fed to the scanning software, which allows the imprint string to be read directly to the BDF as the document control number. Because the Image Address Seed string overwrites the Imprint string, there is no need to assign string segments to the Imprint string. Do not select this check box if you want to use the imprint string set in Scan Designer.

16. In the **Display Image Frequency** field, select which images Scan Operator displays during the scanning process. For example, set this value to 1 to see every image scanned or set it to 2 to see every other image. If you do not want to view any scanned images, set this value to 0. The default value is 1.

17. Select **Save** on the Scanner Configuration Designer window toolbar to save your changes.

Defining Advanced Scanner Settings

To set advanced scanner settings:

1. From FormWare Administration, select the **Scan Designer** button. The Scanner Configuration Designer window displays.

2. Select the Scanner icon on the Config panel. The Scanner Settings tab displays.

3. To use advanced settings, select the **Enable** check box in the Advanced Settings area.

4. Select the **Advanced Settings** button. The scanner-specific Advanced Settings dialogs display for the following scanners:

   Refer to your scanner documentation and the Kofax ImageControls documentation for more information on the scanner specific dialogs. For example, if you have a Bell + Howell scanner, the following dialog displays:
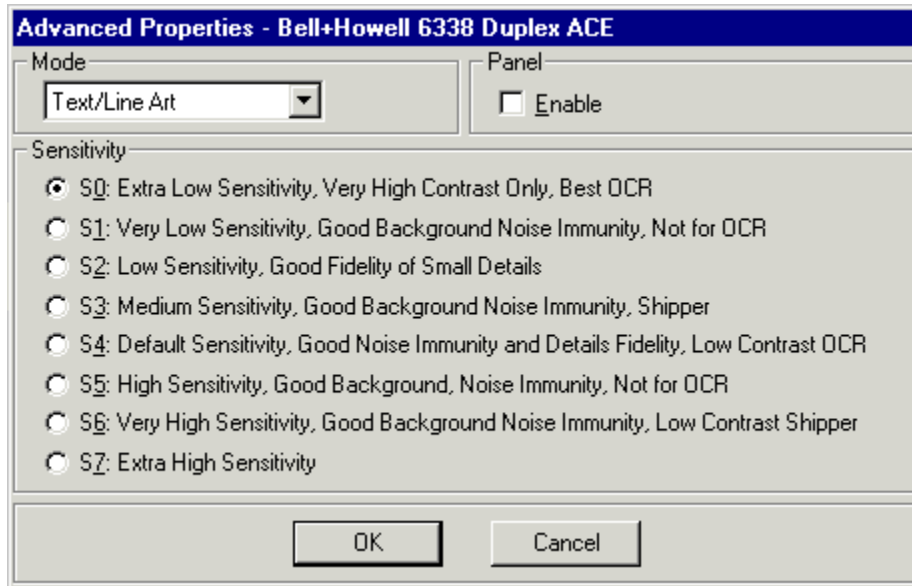
If you do not have one of these types of scanners, the Select Source dialog displays, where you can select a valid scanner source.

5. Select the appropriate settings and then select **OK**.

6. Select **Save** on the Scanner Configuration Settings dialog toolbar to save your changes before you exit Scan Designer.

Defining Image Processing Settings

You define image processing settings to change the way your scanner hardware processes images for viewing on your workstation. You can use these settings to rotate images and select portions of images to reduce image file size and minimize disk space needed for image storage.

To set image processing settings:

1. From FormWare Administration, select the **Scan Designer** button. The Scanner Configuration Designer window displays.

2. Select the Scanner icon on the Config panel and then the Image Processing Settings tab.
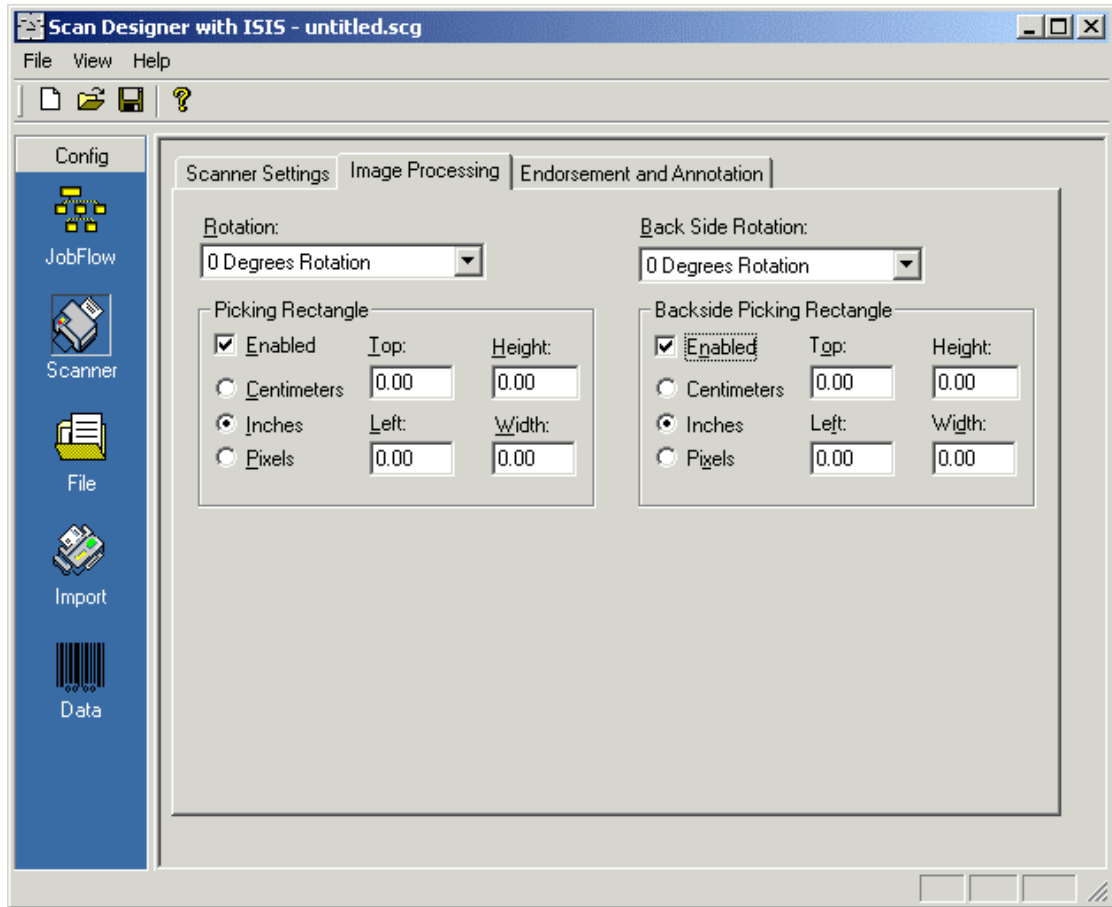
3. In the **Rotation** field, select a value to rotate images clockwise the specified number of degrees, after the image is scanned, but before it is saved to disk. The default setting for Rotation is 0 Degrees Rotation. You might use this value if you are scanning documents with perforated edges, which can cause document misfeeds and jams. You can prevent this problem by feeding a non-perforated edge into the scanner and using this setting to rotate the image. However, this may add time to the scanning process.

4. In the Picking Rectangle area, select the **Enabled** check box to save only the portion of the image in memory, rather than the entire image. This feature enables you to reduce the size of the final image, so that it takes up less disk space. Once enabled, set the following values:

    a. Select **Centimeters**, **Inches** or **Pixels** to set the unit of measure type for the picking rectangle dimensions. If you select **Pixels**, the number of pixels per inch is the same as the dots per inch (DPI). Therefore, a six-inch width on a 200-DPI image would be 1200 pixels.

    b. In the **Top** field, type the value that indicates how far from the top edge of the image the picking rectangle should start.

    c. In the **Height** field, type the value that indicates the distance between the **Top** value and the bottom of the picking rectangle.

    d. In the **Left** field, type the value that indicates how far from the left edge of the image the picking rectangle should start.

    e. In the **Width** field, type the value that indicates the distance between the **Left** value and the right of the picking rectangle.

5. In the **Back Side Rotation** field in the Backside Picking Rectangle area, select a value to flip the backside image clockwise to the specified rotation angle, after the image is scanned but before it is saved to disk. Use this option if the backside of your duplexed images automatically scan upside down; this will rotate the backside images without rotating the frontside images. This field is enabled only if you select **Duplex** as the Scan option on the Scanner Settings tab.

6. Select the **Enabled** check box. These options enable you to save only the portion of the backside image in memory, rather than the entire image. The Backside Picking Rectangle options affect the image from the back of a duplexed page and enable you to save only a portion of the image into memory, rather than the entire image. Once enabled, set the values just as you did in the Picking Rectangle area.

7. Select **Save** on the Scanner Configuration Settings dialog toolbar to save your changes before you exit Scan Designer.

Defining Endorsement and Annotation Settings

You can use Scan Designer to set endorsement and annotation values. Endorsement is the printing of a string of text to a hard copy form by the scanner and annotation is the addition of a string of text to the scanned, binary, image (or TIFF file). Endorsement and annotation occur only during a scanning operation, not during an import operation. ISIS always uses automatic endorsement (i.e., the scanner gets its starting value and starting text from Scan, then the scanner controls the increment). ISIS always uses manual annotation (i.e., Scan generates each piece of text that gets annotated onto the image).

Before you set endorsement and annotation values, verify that your scanner hardware supports these functions or your settings will be ignored. FormWare does not check your scanner hardware to determine whether these options are available.

To set endorsement and annotation values:

1. From FormWare Administration, select the **Scan Designer** button. The Scanner Configuration Designer window displays.

2. Select the Scanner icon on the Config panel and then the Endorsement and Annotation Settings tab.
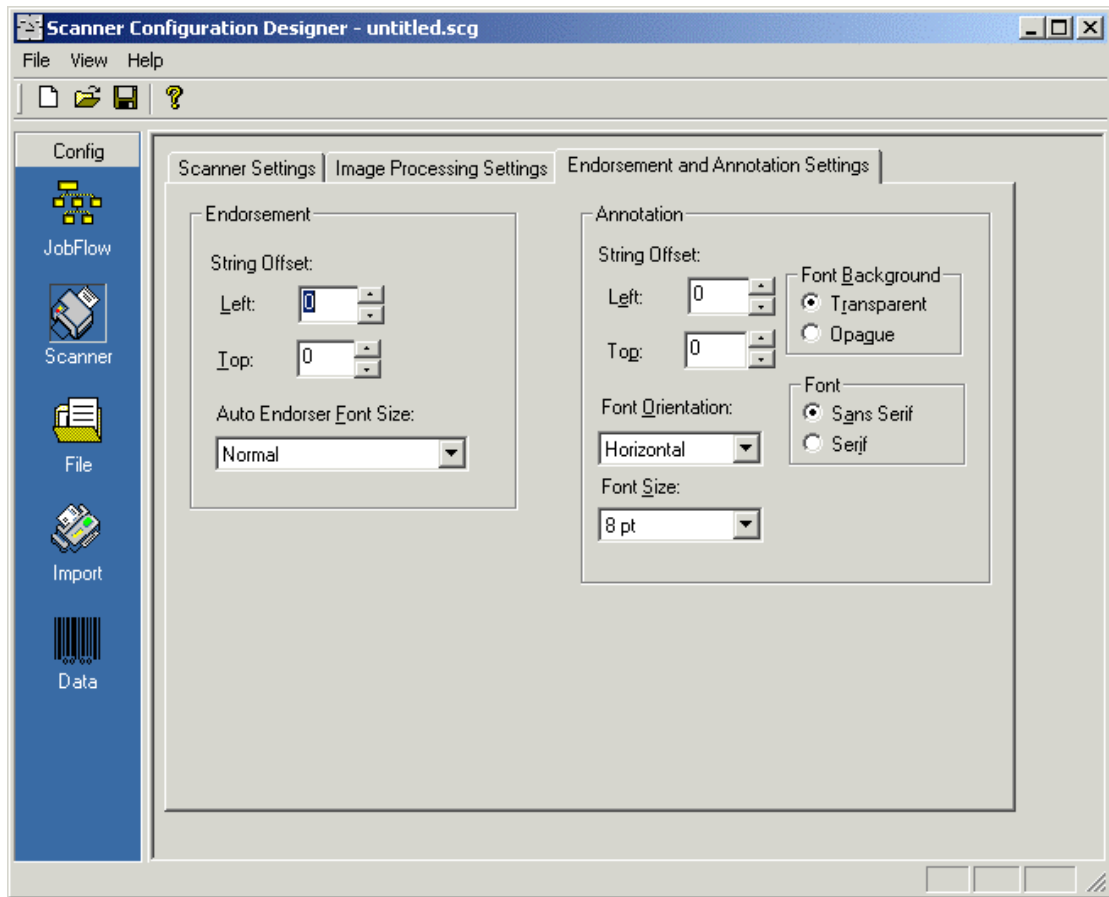
3. In the Endorsement area, set the String Offset values, which indicate the location on your documents where the text string will be printed:

    a. In the **Left** field, select the number of pixels that indicates the distance from the left edge of the document where the endorsement text string will be printed.

    b. In the **Top** field, select the number of pixels that indicates the distance from the top edge of the document where the endorsement text string will be printed.

4. In the **Auto Endorser Font Size** field, select a font size for the endorser on your scanner. Actual font sizes are scanner-specific and not all scanners are capable of using this setting. Check your scanner documentation for more information.

5. In the Annotation area set the following values to indicate the location on your images where the text string will be added and the font characteristics:

    a. In the String Offset fields, set the following values:
        o   **Left** - Select the number of pixels that indicates the distance from the left edge of the document where the endorsement text string will be printed.
        o   **Top** - Select the number of pixels that indicates the distance from the top edge of the document where the endorsement text string will be printed.

    b. In the **Font Orientation** field, select the direction in which FormWare will place the annotation text string on your images. These settings are scanner specific. Valid values are:
        o   **Horizontal** - Places the annotation text string on images horizontally, with zero degrees rotation. This is the default.

o **Vertical** - Places the annotation text string on images vertically, with zero degrees rotation.

c. In the **Font Size** field, select the point size of font used in the annotation text string. The default is 8 pt.

d. In the Font Background area, select the background of your choice by clicking on the appropriate radio button.

o **Transparent** - Allow the background of the image to show through. This is the default.

o **Opaque** - Cover the underlying layer of the image.

e. (Scan over Kofax only) In the Font DPI area, select the Dots Per Inch (DPI) for the font of the annotation text string FormWare places on your images. DPI is the number of dots used within an inch of space on the image. Select **200** or **300** DPI; 200 is the default value.

f. In the Font area, select the font of the annotation text string that FormWare places on your image. Select **Sans Serif** or **Serif**; the default is Sans Serif.

6. Select **Save** on the Scanner Configuration Settings dialog toolbar to save your changes before you exit Scan Designer.

Setting File Properties for Scanning

Using Scan Designer you define the text strings that specify the name and location of the BDF and image files created during scanning or importing or give transaction numbers to a group of BDFs or images. You can also control how the names are incremented or updated as files are created. This functionality ensures that each file is assigned a unique name.

You define text strings in two parts. First, set the prefix string options to specify the full path to the directory where the BDFs or images are to be placed, or the base name of the imprint number. Second, set the string segments to specify how the BDF, image, and imprint string segments are incremented when those types of files are created.

Setting File Properties

To set file properties:

1. From FormWare Administration, select the **Scan Designer** button. The Scanner Configuration Designer window displays.

2. Select the File icon on the Config panel to display the File window.
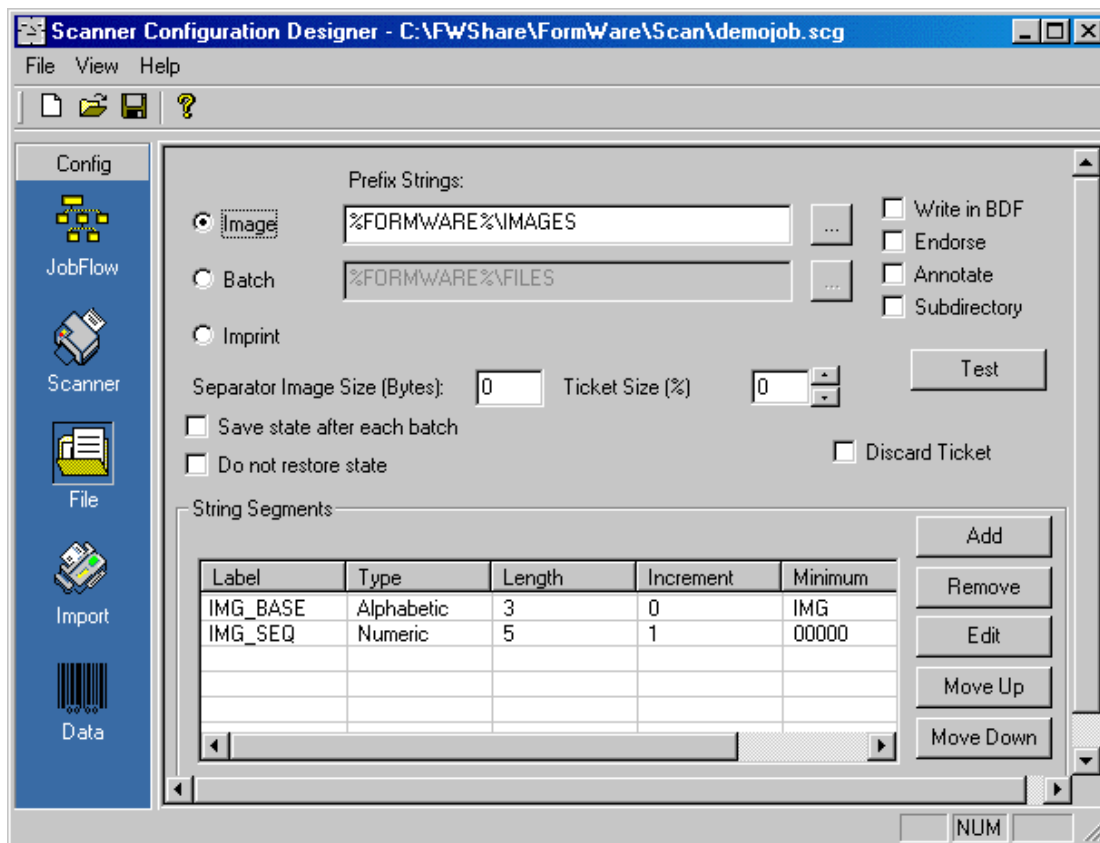
3. Select the Prefix String type from the following options:

    - **Image** – Set a prefix string for each image. In the adjacent field, select the **Browse** button to browse for a folder or type the path to where imported or scanned images will be placed.

    - **Batch** – Set a prefix string for each batch. In the adjacent field, select the **Browse** button to browse for a folder or type the path to where BDFs will be placed.

    - **Imprint** – Indicate the base name of the imprint string segment during a scan or import session using the current configuration. Imprint string segments are also known as Document Control Numbers (DCNs) or Transaction Control Numbers (TCNs).

    Only one Prefix String type can be active at a time on the File properties page; however, you can configure all three strings within a single configuration file. Image and batch strings are required for each Scan Configuration file (SCG), but an imprint string is not. The imprint string does not contain a path name to a file location.

4. In the Imprint String Options area, select the option that designates how your imprint strings will be used if you selected the Imprint prefix string type. The options are *not* mutually exclusive; you can apply any combination of options to the imprint string. Options include:

    - **Write in BDF** - Writes the imprint string value into the BDF during a scan or import session.

    - **Endorse** - Prints the imprint string value on the hard copy document if your scanner supports endorsing.

    - **Annotate** - Writes the imprint string value onto the image (or TIFF file) if your scanner supports annotation. The FormWare Scan annotation library matches the annotation resolution to the image resolution.

- **Subdirectory** - Places the image files into subdirectories based on the BDF name and appends the batch string segment to the image path. Each subdirectory contains the images within a single BDF, which provides for easier file management if you have multiple BDFs. This option is available even if you do not select the Imprint prefix string type.

Note: Although the **Write in BDF**, **Endorse**, and **Annotate** options appear active and can be checked when you have selected the Image or Batch prefix string types, they are not applicable to those types.

---

5. In the **Separator Image Size (Bytes)** field, type the approximate number of bytes in the compressed image of the page minus the header. This is usually between 500 and 1,000 bytes. This allows you to use separators during scanning, but not scan the separators since they will be blank images. The Scan Operator considers all blank pages smaller than this size to be separators and discards them.

Note: You can create blank images during scanning by setting this value to a number greater than zero, e.g., 500-1000 (the number of bytes in the compressed image of the page, which is the TIFF file size minus the header, which is about 350 bytes). All blank pages smaller than this value will be considered "separators" and automatically discarded by the Scan Operator component. This works with ImageControls 3.23, 3.3 and 3.65, but not ImageControls 3.25 (the Kodak specific version). ImageControls 3.65 supports all scanners.

---

6. In the **Ticket Size (%)** field, select the percentage the ticket size is as compared to a regular document size. This option allows you to distinguish between tickets used as batch separators and regular pages during the scanning process. Tickets are either larger or smaller than regular pages and contain batch information such as the batch name, date, count, and other information. The ticket size is set as a percent of the size of the regular pages. For example, if regular pages are 8 ½ by 11 inches and tickets are 8 ½ by 5 ½ inches, the tickets are one-half the size of the pages. Therefore, this value should be 50.

7. Select the **Save state after each batch** check box if you want Scan Operator to save the state file after it finishes scanning each batch. The state file is located in the \FormWare\Tmp directory.

8. Select the **Do not restore state** check box if you do not want Scan Operator to use the state file when you restart after stopping a scanning session in the middle. If you stop a scanning session in the middle, Scan Operator continues with the correct image, batch, and imprint string when you restart it. This is accomplished using a state file in the \FormWare\Tmp directory, which saves information about the state of each machine running the Scan Operator component.

9. Select the **Discard Ticket** check box if you do not want the image of the ticket included in your BDF. Tickets used as batch separators often contain batch information such as name, date, count, and other information. The scanning process appends the image of the ticket and any information to its associated BDF.

10. Select **Save** on the Scanner Configuration Designer window toolbar to save your changes.

Creating String Segments

When you do not assign update properties to a text string, it increments according to the default behavior for a text string. For a text string without a numeric extension, the default behavior assumes that the characters of the name are alphanumeric and updates them from right to left. The text string stops incrementing, or fails to increment, when it reaches the maximum value.

However, if the text string has a numeric extension, the default behavior increments only the extension as a numeric property. The increments also move from right to left. You can modify the default behavior by applying events to the text strings (e.g., when a new file name is created, when another property updates, etc.).

String segments increment one at a time in the order they display in the String Segments table on the File window of the Scanner Configuration Designer dialog.
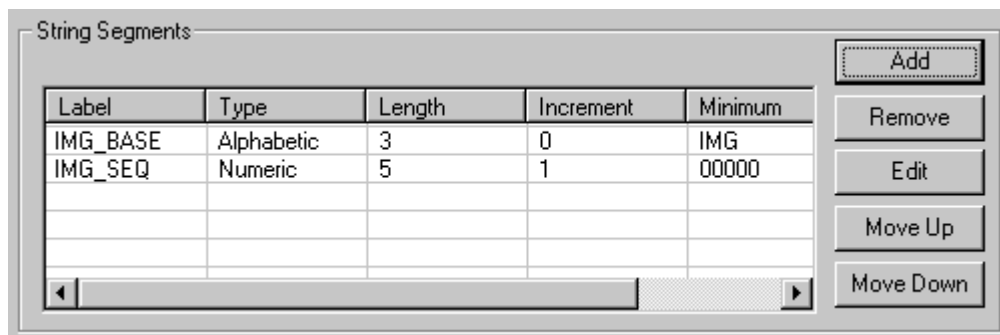
When only one string segment is assigned to a text string, it updates as defined by the properties set. When you assign a second string segment, it increments in response to changes in the first string segment, and so on. If the first string segment updates successfully, all of the following string segments reset to their minimum values. If the first string segment fails to update, the second string segment will attempt to update, if it fails then the third string segment attempts to update, and so on. When none of the string segments can update, the updating of the text string fails and an error message displays.

For example, if you set two string segments to a text string: Julian Date and Numeric and the BDF name starts at 123001.BDF. The Julian Date string segment is listed first and therefore is the first to attempt to update. The behavior of the second, Numeric, string segment is dependent upon the update behavior of the first field. With these settings, the file name updates from 123001.BDF to 123002.BDF to 123003.BDF. The Julian Date string segment first attempts to update but fails (since the date has not changed), so the Numeric string segment updates. When the date changes, the Numeric string segment resets to 124001.BDF.

The **Move Up** and **Move Down** buttons change the order in which the string segments display, and thus the order in which the string segments attempt to update. In the example above, if the display of the Numeric and Julian Date string segments are reversed, the filename displays as 001123.BDF. Each time a new document is scanned or a new image is imported, the Numeric string segment will update. Because the Numeric string segment updates every time, the Julian Date string segment will never update.

In the String Segments area, you control how the names of BDFs, imprint strings, and image files are incremented to ensure unique names. You can select from a variety of naming mechanisms. Image and batch names are limited to 27 characters, with a maximum of nine characters per string segment.

To create a string segment:

1. Select the **Add** button on the File window of the Scanner Configuration Designer dialog. The String Segment Properties dialog displays.

2. In the **Label** field, type the unique name assigned to the string segment. This name is referenced when linking string segments for incrementing. Because incrementing behaviors can be linked to other string segments within all three strings, each label must be unique within the SCG file.

3. In the **Type** field, select the data type of the string segment, or the limitation of the kinds of characters that can exist in the string segment. Valid data types are:

   - **Alphabetic** - Applies the alphabetic updating property to the selected characters in the file name.

   - **Alphanumeric** - Applies the alphanumeric updating property to the selected characters in the file name.

   - **Numeric** - Applies the numeric updating property to the string segment.

   - **Date** - Applies the date updating property to the selected characters in the string segment. You can modify this character set to change the format of the date displayed.

   - **Time** - Applies the time updating property to the string segment. The time value for this property is calculated as military time. You can modify this character set to change the format of the time displayed.

   - **Julian Date** - Applies the Julian date updating property to the selected characters in the file name. The Julian date is the count of calendar days. For example, January 1 is 001, February 3 is 034, etc. You can modify this character set to change the format of the Julian date displayed.

   - **Custom** - Applies the custom updating property to the characters in the string segment. When you select Custom, you can limit the character set as well as determine a user-defined increment order using alphanumeric characters. When you select this option, the **Character Set/Format** field is enabled so you can type a custom character set. For example, if you type "1234ABCD5678EFGH" as the custom character set, and that string segment is the only one applied to the text string, the "00000003.TIF" file name will update to "00000004.TIF", and then "0000000A.TIF", etc.

   - **Backslash** - Adds a backslash to the text string. This only applies to the Label.

   - **Bar Code** - Updates the string segment when the specified bar code type is identified during the scanning process.

4. In the **Increment** field, select the value that represents the units by which the string segment will be incremented. For example, if you select Numeric in the **Type** field and want the string

segment to update by an increment of five, type 5 in this field. The first value in the string segment would be zero, the second would be five, the third would be ten, etc.

5. In the **Character Set/Format** field, type or modify the character set values. If you selected Custom in the **Type** field, you can type any alphanumeric values in this field; however, they are changed to uppercase. If you selected Date, Time, Julian Date, or Custom in the **Type** field, you can edit the character set values. You can type any alphanumeric data in the order you require. However, remember that edited character sets affect the storage space available for the updating properties of a file name.

6. In the **Minimum** field, type the lowest, or initial, value the string segment can reach. When the string segment resets, it returns to this value.

7. In the **Maximum** field, type the highest value that the string segment can reach. When the string segment reaches this value, the next value returned will be the number in the **Minimum** field.

8. In the **Value** field, type the first value the string segment will output. String segments cannot overlap characters of another string segment.

Note: The value in the **Value** field must be greater than zero for the Panasonic 905c scanner.

---

9. Select the **Edit on Start** check box to make a string segment available for editing at the time you select an SCG file in Scan Operator. When selected, the operator can redefine the contents of the field before the scanning operation. In this case, when the scanning operation starts, the Edit on Start dialog displays the label and an edit box with the current field value. The operator can change the field value for the current scanning operation.

10. Select the **Events** button to add an event for the string segment (see Creating Events for String Segments) or **OK** to return to the File window.

11. Use the **Remove, Edit, Move Up,** and **Move Down** buttons to modify and rearrange your string segments.

12. Select **Save** on the Scanner Configuration Designer window toolbar to save your changes.

Creating Events for String Segments

You can create events that cause a string to increment or reset. To do this:

1. From the String Segment Properties dialog, select the **Events** button. The Events dialog displays.
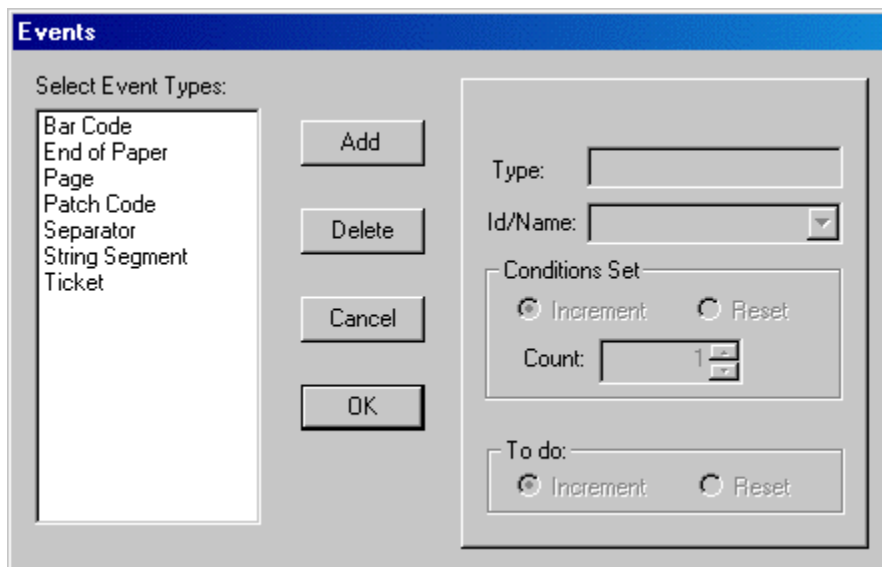
Note: You cannot add events to a Time, Date, Julian, Backslash, or Bar Code string segment.

---

2.   Highlight an event type and select the **Add** button. Valid event types are:

- **Bar Code** - Indicates the string segment updates when the specified bar code is identified during the scanning process. Before assigning a bar code event, you must enable bar code and add a bar code type to batch processing on the Bar Code tab of the Data window in the Scanner Configuration Designer dialog. Then, assign the bar code type by selecting it from the **ID/Name** field on this dialog.

- **End of Paper** - Indicates the string segment updates when the scanner reaches the end of the paper.

- **Page** - When used with a page-level string segment, the page event indicates the string segment updates when a new image is created by the scanner and added to the BDF. When used with a batch-level string segment, the page event creates a new BDF on the page count you set.

- **Patch Code** - Indicates the string segment updates when the specified patch code is identified during the scanning process. Before assigning a patch code event, you must enable patch code and add a patch code type to batch processing on the Patch Code tab of the Data window on the Scanner Configuration Designer dialog. Then, assign the patch code type by selecting it from the **ID/Name** field on this dialog.

- **Separator** - Use separator pages to determine the number of images per batch. The separator event provides a more accurate means of control for creating BDFs. The separator page must be the first document scanned, the same size as the other documents you are scanning, and a blank page. When the scanner encounters a blank page, it creates a new BDF and discards the separator page. You can also set a maximum separator file size to determine a separator page. To set the maximum separator file size, type the maximum number of bytes you want a separator to contain in the **Separator Image Size** field on the File window. If the size of a scanned image is less than or equal to this value, the system treats the image as a separator. If the **Separator Image Size** is set to zero, FormWare compares the size of each image to the first image scanned. If the byte size of an image is within 5,000 bytes of the first image, it is treated as a separator.

- **String Segment** - Displays the list of all string segment labels within the current scan configuration. Corresponding Increment/Reset options also display, to indicate whether

---

the current string segment should update when the event string segment increments or resets.

- **Ticket** - Use tickets to determine the number of images per batch to provide a more accurate means of control when creating BDFs. The ticket document must be at least 20% shorter or longer than the other documents being scanned. The batch tickets you use within a single scanning process must all be the same size. Set the ticket size by typing a number in the **Ticket Size** field on the File window. The ticket must be the first document scanned within each batch. Each time the scanner encounters a ticket during the scanning process, it creates a new BDF with the ticket image as the first image in the BDF. If you do not want tickets included in the BDF, select the **Discard Ticket** check box on the File window.

Note: The ticket event requires that you have a scanner with an Auto Length Detection (ALD) feature. Without ALD, each scanned image is padded out to the defined page size (e.g., 8.5 x 11). Because the ticket size cannot be distinguished from the actual document size, the ticket mode cannot function. With ALD, there is no automatic padding of images. Refer to the literature that accompanied your scanner to determine if your scanner supports this feature.

You can switch between event tabs by selecting the numbered tabs. To delete an event, select the tab for the event you want to delete, and select the **Delete** button.

Set values in the following fields as appropriate for the event type you are creating. Depending upon the type of event you selected, some fields will not be available. The event type you selected displays in the **Type** field.

3. In the **ID/Name** field, indicate the Bar Code, Patch Code, or String Segment type to be used by the event. This option is available only for the Bar Code, Patch Code, and String Segment event types. You must set properties on the Bar Code and Patch Code tabs of the Data window on the Scanner Configuration Designer dialog before you can create a Bar Code or Patch Code event here.

4. Set the options in the Conditions Set area depending upon the event type you selected. This area does not apply to the Bar Code, End of Paper, Patch Code, Separator, and Ticket event types.

- For a Page event type in the **Count** field, select the number of scanned or imported images allowed in a batch before a new BDF is created. For example, if you set the batch Count to 100, the document Count to 5, and scan 20,000 images, then 200 BDFs are created, each containing 20 documents. Each of those documents contains 5 images.

- For a String Segment event type, select **Increment** or **Reset** in the Conditions Set area to determine how the string segment for which the event is created will react to changes in another string segment. Selecting **Increment** or **Reset** in the To do area defines how you want the string segment to react to the change in the controlling string segment.

5. In the To do area, select from the following options:

- **Increment** - Enables you to select the value that represents the units to increment the string segment. This option is available in the To Do area of all event panels.

- **Reset** - Resets the string segment to the lowest possible value for the property type. Use this option when you want the field to reset to the minimum value. For example, a numeric string segment resets to 0000 and an alphabetic or alphanumeric field resets to AAAA. If you do not select this option, the string segment resets to 0001 or AAAB.

Note: Test all events you set for infinite looping logic. You test events using the **Test** button on the File window.

6. Select **OK** to save the events you defined and return to the String Segment Properties dialog.

7. Select **OK** to return to the File window of the Scanner Configuration Designer dialog.

8. Select **Save** on the Scanner Configuration Designer window to save your file settings.

Testing String Segment Updates

You should always test your string segments before using them in production to ensure they update correctly. To test your text strings:

1. On the Scanner Configuration Designer dialog, File window, select the **Test** button. The Edit on Start dialog displays the string segments assigned to the Image String. Select the **OK** button or change the text in a **Value** field and then select **OK**.

2. If you did not select the **Edit on Start** check box for any of your string segments, the File String Test dialog displays.
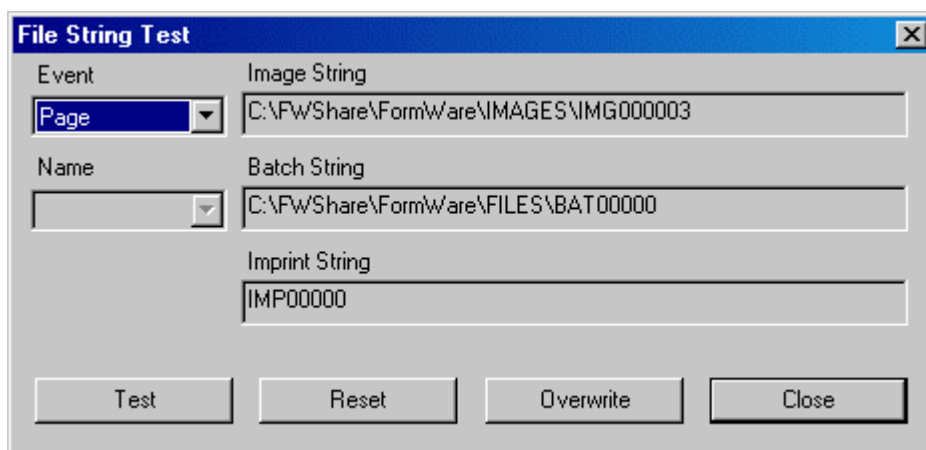
You can test the following features of your test strings on the File String Test dialog:

- Image and BDF names incrementing.
- Directory names and structure changes.
- Imprint string changes.
- Segment updating and linking.

Note: With the exception of the **Overwrite** feature, this testing mechanism does not impact or change the current settings and properties. Therefore, unless you select the **Overwrite** button, you will not need to reset the settings and properties of the SCG file after testing unless the testing reveals a problem.

3. From the **Event** field, select the event type you want to test and select the **Test** button to see how that event type updates your Image, Batch, and Imprint Strings. For example, set a numeric string segment with an initial value of 001 to update on each Page event, select Page in the **Event** field, and select the **Test** button. The string segment updates to 002.

4. Select a string segment in the **Name** field and select the **Test** button to determine how the event updates the string segments. The list contains the string segments that can be associated with the String event. This option is not valid until you have selected an event type and have set string segment events for one or more of the string segments attached to the text strings.

5. Select the **Overwrite** button to change the value of your string segment to the number in the **Batch String** field. As a result, when you begin processing, the first value used will be the value in the **Batch String** field when you selected **Overwrite**.

6. Select the **Reset** button to return the string segments on this dialog to their original values.

7. Select **Close** to end your test and return to the File window.

8. Select **Save** on the Scanner Configuration Designer dialog toolbar to save your changes.

Setting Scanner Mode for Duplex Scanning

To take advantage of improved endorsement when scanning duplex, set your batch increment to an even number and double the number of documents you expect in the batch. FormWare counts the front and back of each page when scanning duplex, so if you scan two documents, the result will be four images, and FormWare would increment to four. For example, to duplex scan a batch that contains three documents you must set your batch increment to six. Additionally, you must set the imprint string segment to a page count of two. You make these settings on the Scan Designer File tab. Note the number of pages in your batch and then set your scanner for duplex.

To set your batch increment:

1. From Scan Designer, select the File icon.

2. Select the **Batch** option.

3. In the String Segments area, select the Numeric string segment and select the **Edit** button. The String Segment Properties dialog displays.

4. Select the **Events** button.

5. Select the tab with Page in the **Type** field.

6. In the **Count** field in the Conditions Set area, select the number that is double the page count in your batch (e.g., select 6 for a three-page batch).

7. Select **OK** on the Events dialog.

8. Select **OK** on the String Segment Properties dialog.

To set the imprint sting segment:

1. From the File tab, select the **Imprint** option.

2. In the String Segments area, select the Numeric string segment and select the **Edit** button. The String Segment Properties dialog displays.

3. Select the **Events** button.

4. Select the tab with Page in the **Type** field.

5. In the **Count** field in the Conditions Set area, select 2 for duplex scanning.

6. Select **OK** on the Events dialog.

7. Select **OK** on the String Segment Properties dialog.

Setting Up Auto Endorsement for Duplex Scanning

FormWare supports auto endorsement and auto annotation only on scanners capable of those automatic processes. The following procedure is for using auto endorsement with duplex scanning:

1. Run Scan Designer.

2. On the File tab, select the **Batch** option and the **Endorse** check box.

3. Count the documents to be scanned and place them in the scanner tray.

4. In the String Segments area, select the desired string segment and select the **Edit** button.

5. Select the **Events** button.

6. Select **Page** and in the Select Event Types area and select **Add** to create a Page tab**.**

7. On the Page tab change the Count to double the number of documents in the tray (if ten documents are on the scanner, the page count must be 20 or more—always an even number).

8. Select **OK** twice.

9. To keep data written in the BDF synchronized with the data imprinted/endorsed on the hard copy document, select the **Imprint** option and then select your batch in the String Segments area.

10. Select **Edit** and then **Events.**

11. On the Page tab, change the **Count** field value in the Conditions Set area to 2.

12. Select **OK** twice.

Setting Scanner Mode Imprint Strings for Kodak Scanners

The Scanner Mode options available on the Scanner Settings dialog within Scan Designer allow you to use the Imprint string supported by some Kodak scanners. For these settings to be effective, you must create Imprint string segments that correspond to the Used segments and Separator configurations selected on the File Properties dialog.

Caution: Use the Scanner Mode options only with Kodak scanners.

You can configure the Imprint string either from Scan Designer or from the scanner's LCD panel. The Scan process updates the Imprint string according to information fed to the system by the scanner. The Imprint string, which is sent from the scanner to FormWare, is also referred to as the Scanner Derived String. It can overwrite the Imprint string and then be written into the BDF. This means that the information originating from the scanner and printed on the hard copy document can match the information written into the BDF as well as the information annotated on the image file.

When you configure an Imprint string in the Scan Designer module, the Imprint string is passed to the scanner to serve as the string that the scanner begins with for the current Preset mode. Below is an example of an Imprint string created in Scan Designer for use with the Used segments configuration F-3-1 and the period (.) Separator.

The Imprint string and its string segments can have up to 15 characters, including the periods that separate the fields within the string. Enter a character in the Imprint string segments on the File window to represent each character within a Used segment. For example, if you want Used segment 3 to have three characters, then specify "000" in the related string segment.

Note: The Imprint string and its string segments can have up to 40 characters if you use a 3500 series Kodak scanner.

The Imprint string configured in Scan Designer must have a structure corresponding to the Scanner Derived String specified on the Kodak scanner control panel, That is, if you use a 3.6.2 string, 000.000000.00, then the same string structure must be defined on the File window of the

Scanner Configuration Designer dialog. You can set the Imprint string to a different character set (e.g., ABC.000001.22), but the structure must match or you will receive an error message.

Test the way your Imprint string increments to ensure that it updates correctly. Perform this test using the File String Test dialog.

Setting Import Properties for Scanning (ISIS)

Import properties define how you import images from a scanner, a fax, or another directory into the FormWare directory structure. You define these settings in Scan Designer and they are used by the Scan Operator component.

The Source Directory is the directory from which files are imported. Image files are always imported to the directory designated in the Image String field on the File window within Scan Designer no matter whether the images are imported, scanned, or faxed. When you create BDFs during an import operation, they are imported into the directory specified in the Batch String field of the File window. Note that the destination directory of the image files is different from that of the BDFs. You can edit or view the image and batch strings from the File window.

When the Source Directory is different from the Destination Directory designated by the image string, an import operation moves files from the Source Directory to the Destination Directory. Only those files that match the pattern you define are moved. When the Source and Destination Directories are the same, no files are moved. When you import faxes, the Destination Directory must be different from the Source Directory.

To set import properties:

1. From Administration, select the **Scan Designer** button to display the Scanner Configuration Designer window.

2. Select the Import icon on the Config panel.
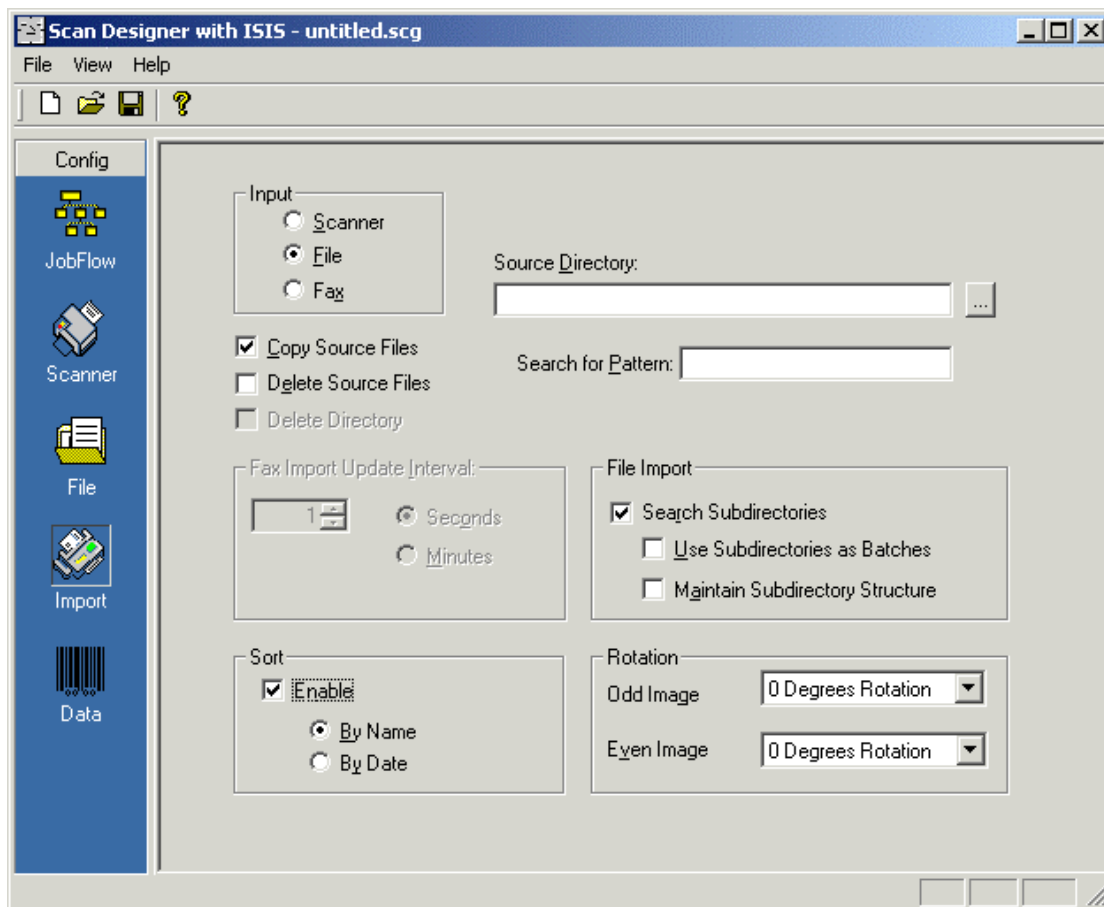
3. In the Input area, select one of the following options to indicate the source of the files to be imported into FormWare. Your selection here activates certain fields on this window. Options are:

    - **Scanner** – Import images using a scanner. If you select this option, all remaining options on this window are disabled.

    - **File** - Specify properties for importing image files into a BDF. Image files can keep their original names or be renamed to conform to an existing naming convention.

    - **Fax** - Monitor incoming faxes and then import those faxes at a specific time interval. If you receive forms via fax, you can still use the Scan Designer module to create BDF files. Many fax servers are able to create images in a specified folder. Most fax servers create images using a Group3 compression scheme. These images do not have a TIFF image header, which contains image tags that hold information about the image. Without this header, some recognition and image enhancement software may function incorrectly. For this reason, the Scan Designer module checks to see if the image contains a TIFF header and automatically adds one during the importing process.

4. In the **Source Directory** field, select the adjacent button to browse for the directory path to the folder that contains image files to import. This must be an existing folder.

5. In the **Search for Pattern** field, type a pattern to limit the import function to only file names with this pattern. File names that do not match this string are not imported. You can use wild card (*) characters in your search to import a range of files from within the specified source directory.

6. Select the **Copy Source Files** check box to copy files from their original location during import.

7. Select the **Delete Source Files** check box to delete files from their original location during import. This check box is only available if you select the **Copy Source Files** check box.

8. Select the **Delete Directory** check box if you have selected the **Delete Source Files** check box to delete subfolders when importing images.

9. In the Fax Import Update Interval area, select the numeric value that indicates the time interval the Scan Designer module will poll the Source Directory to determine if a new fax image file exists. Select either **Seconds** or **Minutes** to determine whether the value is in seconds or minutes. The default is one second. This option is active only when you have selected **Fax** in the Input area.

10. In the File Import area, select the **Search Subdirectories** check box to import images from the subdirectories of the directory specified in the **Source Directory** field. This option is active only when you select **File** in the Input area. Selecting this check box activates the following options:

    - **Use Subdirectories as Batches** - Use the subdirectories under the Source Directory to delineate batches. For example, if you specify c:\Images\Project1 as your Source Directory and it contains four subdirectories (GroupA, GroupB, GroupC, and GroupD), then four batches are created using the subdirectory names (i.e., GroupA.bdf, GroupB.bdf, etc.).

    - **Maintain Subdirectory Structure** - Maintain the subdirectory structure, folder names, and images they contain during the importing process. When images under the Source Directory are moved to the Destination Directory, they stay within their current subdirectory structure, contain the original images, and maintain their original names. When you select this option, you do not need to set any string segments to update the file names. If you do not select this option, the new subdirectories are created with names based on the BDF file name, specified in the **Batch String** field on the File window.

11. In the **Sort** area, select the **Enable** check box to enable the sort options. Options are:

    - **Name** - Sorts imported images in ascending order by file name

    - **Date** - Sorts imported images in ascending order by file date.

12. In the **Rotation** area, select whether to rotate images during import. Select values in the **Odd Image** and **Even Image** fields to rotate images 0, 90, 180, or 270 degrees.

13. Select **Save** on the Scanner Configuration Designer window toolbar to save your changes.

Setting Data Properties for Scanning (ISIS)

Data properties within Scan Designer specify the bar code and patch code options to use for form identification (Form ID) and batch processing during the scanning process. There are two tabs on the Data window:

- **Bar Code** - to specify bar code options for your scanner.

- **Patch Code** - to specify patch code options for your scanner.

These features are available only if they are supported by your scanning hardware. The PixTools/IP bar code engine replaces the Kofax bar code engine.

Defining Bar Code Scan Settings

FormWare Scan with ISIS module will do bar code and patch code recognition only on binary images. Hardware bar codes and patch codes work only with Kodak scanners that support hardware bar code. Other than Kodak, a bar code or a patch code is found through software, not through hardware.

The Bar Code tab is active regardless of whether your scanning hardware supports bar code functionality. Check your scanning hardware documentation to determine whether your scanning hardware supports bar codes and to find out the specific bar code types it supports. You can also enable bar code recognition from an Adrenaline card if your scanner does not support bar code recognition. Refer to the Adrenaline card documentation for more information.

The left side of the Bar Code tab contains options that apply to all bar codes specified and the right side contains tabs, each representing a single bar code. You can specify a number of bar codes to be associated with either the image string or the batch string; however, you will use only one bar code per text string for Form ID during the scanning process.

To set bar code data properties:

1. From Administration, select the **Scan Designer** button. The Scanner Configuration Designer window displays.

2. Select the Data icon on the Config panel. The Bar Code tab displays.
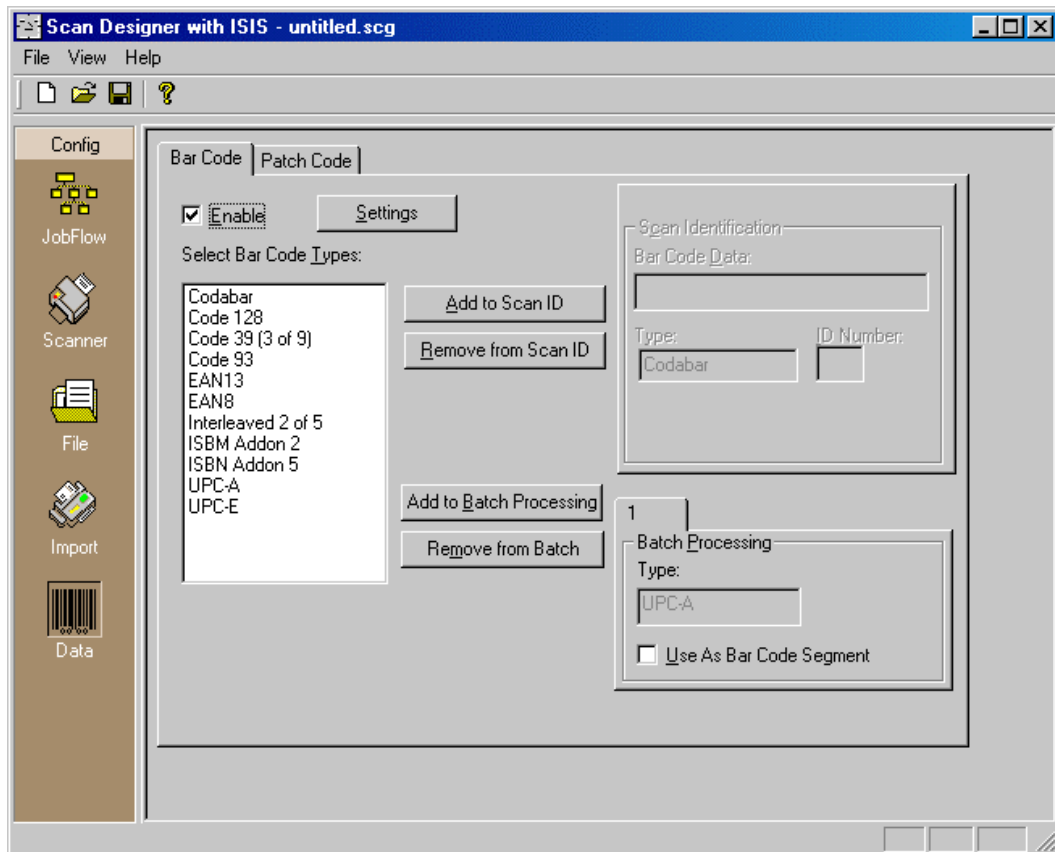
3. Select the **Enable** check box to activate bar code functionality and see the supported bar code types.

4. Highlight a bar code type and select either the **Add to Scan ID** button to add the bar code type to the Scan Identification group or the **Add to Batch Processing** button to add the bar

code type to the Batch Processing group. Add as many bar code types as you want to either group. Select from the following bar code types:

- Codabar - Binary bar code of variable length. The valid character set includes 0–9 and the special characters: / . +. Codabar uses series of wide and narrow elements for character representation and does not support the check sum feature.

- Code 128 - Proportional bar code of variable length. The valid character set includes all 128 ASCII characters. This dense bar code comprises six elements (three bars and three spaces). With the exception of the stop character, all other characters are 11 units long. Check sum is mandatory with the Code 128 bar code.

- Code 39 (3 of 9) - Binary bar code of variable length. The valid character set includes A–Z, 0–9, and the special characters $ - : / . +. The asterisk (*) is used as the start and stop character. Check sum is optional with the Code 39 bar code.

- Code 93 (9 to 3) - Proportional bar code of variable length. The valid character set includes A–Z, 0–9, the space, four shift characters, and the special characters $ - % . + /. This structurally complex bar code requires two check digits. Code 93 does not support the check sum feature.

- EAN-13 and EAN-8 (European Article Numbering) - Fixed length code with either eight or 13 characters. The last digit is a check digit. Check sum is mandatory.

- Interleaved 2 of 5 - Variable-length bar code that uses five black bars and five white bars to define a number. Check sum is optional with this bar code type.

- ISBN Addon2 - The International Standard Book Number (ISBN) is a machine-readable identification number that marks any book unmistakably. The U.S. ISBN Agency assigns the ISBN Publisher Prefix to those publishers with a residence or office in the U.S. and who are publishing their titles within the U.S. The Addon2 and Addon5 codes are a 4-width system (thin, narrow, wide, fat) where each symbol consists of two bars and two spaces, and the total width is always the same. A two-bar separator is between each set of four. There is no start code or stop code and no check sum. They only encode digits.

- ISBN Addon5 - Accompanies a UPC-A/EAN-13 bar code, placed just to the right. Addon5 is often used to specify which size or shape of a particular line of products that otherwise share the same EAN-13 (e.g., postal envelopes and mailers).

- UPC-A (Universal Product Code-A) - Numeric-only, fixed length 12-character bar code, designed to identify a product and its manufacturer. This is a standard bar code scheme for items sold to the general public. Supplementals are supported.

- UPC-E - Numeric-only, zero-suppressed version of UPC-A. UPC-E compresses the data characters and the check sum into six characters.

5. Define the settings for each bar code type you added to the **Scan Identification** group. The Scan Identification process identifies forms during scanning. This feature will match your settings against bar codes on hard copy documents to identify each document's form type. To define the scan ID for a bar code type, select the tab for that type and do the following:

   a. In the **Bar Code Data** field, type the data to be matched against the bar code, which is then used to identify the form. You can use an asterisk (*) as a wild character if the values you type do not match the bar code data exactly.

   b. In the **ID Number** field, type the number associated with all scanned documents that contain this bar code. This number is placed before the corresponding image name in the BDF and is used for form identification.

6. Define the settings for each bar code you added to the Batch Processing group. In Batch Processing, you define the bar codes that will be used to create batch separators. When the

scanning process detects the specified bar code, it generates a new BDF. The bar code type displays in the **Type** field. Select the **Use As Bar Code Segment** check box to assign a bar code as the batch name during the scanning process. This allows you to easily identify a batch based on its bar code. When you select this option, a new string segment is defined that recognizes bar codes. This segment requires the string name generator to use a bar code as the value of the segment.

7. To remove a bar code type from the Scan ID process, select the numbered tab for the bar code type and select the **Remove from Scan ID** button. To remove a bar code type from the Batch Processing group, select the numbered tab for the bar code type and select the **Remove from Batch** button.

8. Select **Settings**. The Barcode Detection - Properties dialog.

9. Select the **All** check box to instruct FormWare to use the scanner hardware to recognize bar codes if the scanner hardware can do so. Your scanner hardware may not recognize all bar code types. Or, uncheck **All** and leave checked only those bar code types that apply to your scanner.

10. In the Symbologies to Check area, the following check boxes (defined previously) are possible:
    - All--detect and decode all listed bar codes
    - Code 3 of 9--Code 39 bar codes
    - Interleaved 2/5--2 of 5 bar codes
    - Code 128--Code 128 bar codes
    - CODABAR--Codabar bar codes
    - UPC-A--UPC-A bar codes
    - UPC-E--UPC-E bar codes
    - EAN-8--EAN-8 bar codes
    - EAN-13--EAN-13 and ISBN bar codes
    - Code 9 of 3--Code 93 bar codes
    - ISBN Addon2--ISBN Addon 2 bar codes
    - ISBN Addon5--ISBN Addon 5 bar codes

11. In the Region Properties area, determine the top and left location and the height and width of the bar code on the image. In the **Left** and **Top** fields, select the location of the bar codes on your hard copy documents in inches. In the **Height** and **Width** fields, select the dimensions of the bar codes on your hard copy documents in inches.

12. In the **Orientation** field, select from the following options:
    - Horizontal Search for horizontal bar codes.
    - Vertical Search for vertical bar codes.
    - Diagonal Search for diagonal bar codes.

13. Select the **Decode Symbol** check box to decode bar code data into human readable ASCII text.

14. Select the **Multiband Analysis** check box to use the multiband analysis detection algorithm. This algorithm is a time-consuming search technique that may increase detection accuracy when used on poor-quality source images.

15. Select the **Exact Length** check box to detect only bar codes with a known character length. Type the number of characters in the expected bar codes in the associated field.

16. Select the **Minimum Height** check box to detect only bar codes that satisfy a minimum height requirement. Type the desired minimum height in the associated field.

    You can use the following units of measure for the **Exact Length** and **Minimum Height** properties:

    - Inches (in or ")
    - Centimeters (cm)
    - Millimeters (mm)
    - Pixels (pi)
    - Points (pt)

    You may specify a value as a fraction (1 1/2 in or 1 1/2"), or as a decimal (3.81 cm, 38.1 mm, or 108.5 pi); however, you must use whole units when using points (108 pt) to express values.

17. Select **OK** and then **Save** to save your settings.

Defining Patch Code Scan Settings

You typically use patch codes as a document separator, batch separator, or form/document identifier. Unlike bar codes, patch codes do not contain data. FormWare/Scan supports Patch Code recognition using a software algorithm. As a result, patch code detection is available in a consistent manner on all scanners regardless of the scanner's capabilities.

The left side of the Patch Code tab lists the three supported patch code types, and the right side contains a scan ID tab for each selected patch code type. You can specify a number of patch codes to be associated with the image string; however, you will use only one patch code per image string for Form ID during scanning.

To set patch code data properties:

1. From Administration, select the **Scan Designer** button. The Scanner Configuration Designer window displays.

2. Select the Data icon on the Config panel.

3. Select the Patch Code tab.
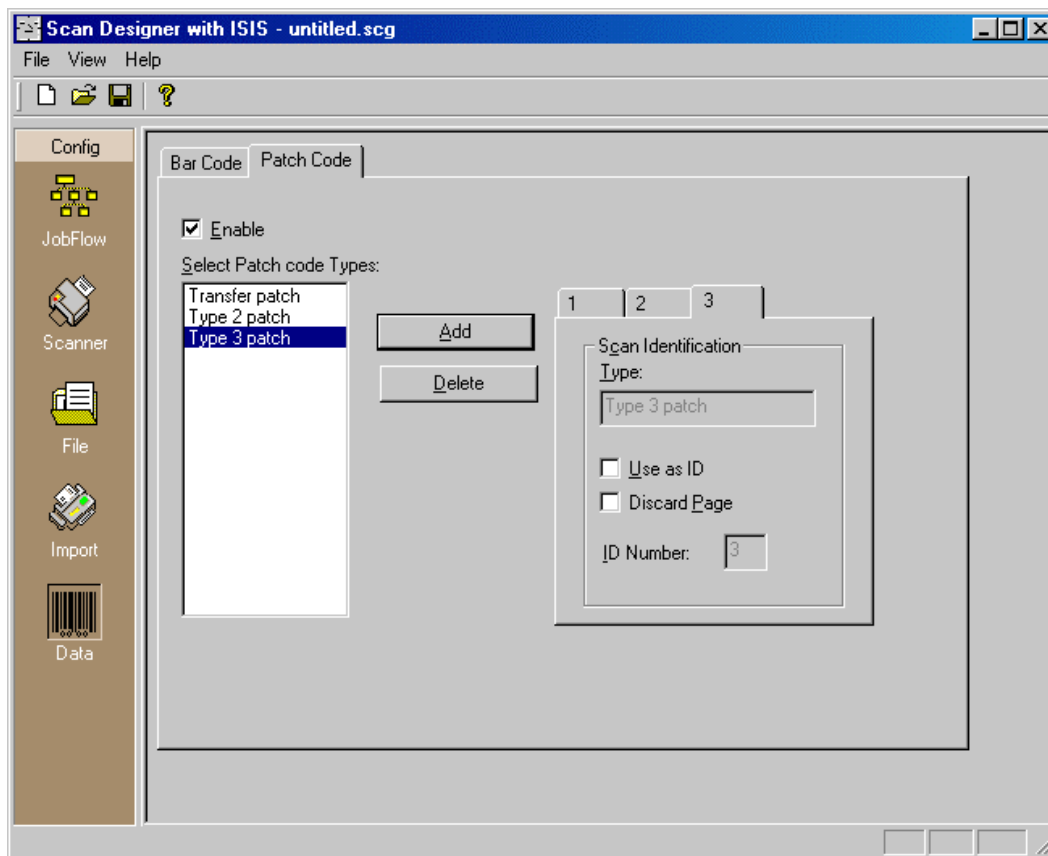
4. Select the **Enable** check box to activate patch code functionality.

5. In the **Select Patch code Types** field, highlight a patch code type and select the **Add** button. The patch code type determines the kind of patch code on the image. You can add as many as three patch code types: Transfer patch, Type 2 patch, and Type 3 patch.

6. In the Scan Identification area, you define the processing settings for the patch code type displayed in the **Type** field. You can set the following options:

    a. Select the **Use as ID** check box to assign a patch code as the batch name during the scanning process. This allows you to easily identify a batch based on its patch code. When you select this option, a new string segment is defined that recognizes patch codes. This segment requires the string name generator to use a patch code as the value of the segment.

    b. Select the **Discard Page** check box to instruct FormWare to discard pages that do not have the correct patch code to use in Form ID.

    c. In the **ID Number** field, you can type the number associated with all scanned documents containing the specified patch code if you have selected the **Use as ID** check box. This number is placed before the corresponding image name in the BDF and is used for Form ID.

7. To delete a patch code type, select the numbered tab for that patch code in the Scan Identification area and select the **Delete** button.

8. Select **Save** on the Scanner Configuration Designer window toolbar to save your changes.

Setting Data Properties for Scanning (Kofax)

If you are using the FormWare Scan with ISIS module, refer to the Setting Data Properties for Scanning (ISIS) section. Refer to this section only if you are using FormWare Scan with Kofax. Data properties within Scan Designer specify the bar code and patch code options to use for form identification (Form ID) and batch processing during the scanning process. There are two tabs on the Data window:

- Bar Code - to specify bar code options for your scanner.
- Patch Code - to specify patch code options for your scanner.

These features are available only if they are supported by your scanning hardware.

Defining Bar Code Scan Settings

The Bar Code tab is active regardless of whether your scanning hardware supports bar code functionality. Check your scanning hardware documentation to determine whether your scanning hardware supports bar codes and the specific bar code types it supports. You can also enable bar code recognition from an Adrenaline card if your scanner does not support bar code recognition. Refer to the documentation for your Adrenaline card for more information.

The left side of the Bar Code tab contains options that apply to all bar codes specified and the right side contains tabs, each representing a single bar code. You can specify a number of bar codes to be associated with either the image string or the batch string; however, you will use only one bar code per text string for Form ID during the scanning process.

To set bar code data properties:

1. From Administration, select the **Scan Designer** button. The Scanner Configuration Designer window displays.

2. Select the Data icon on the Config panel. The Bar Code tab displays.

3. Select the **Enable** check box to activate bar code functionality.

4. Select the **Check Sum** check box to reserve the last character in the bar code for the check sum option. The bar code recognition driver uses this last character to validate the bar code. The program uses the check sum character for validation purposes only and does not return this character as part of the bar code data.

5. Select the **Use Scanner if Possible** check box to instruct FormWare to use the scanner hardware to recognize bar codes if the scanner hardware is able to do so. However, your scanner hardware may not recognize all bar code types.

6. In the Bar Code Size area, determine the height and width of the bar code on the image. In the **Height** and **Width** fields, select the dimensions of the bar codes on your hard copy documents in inches.

7. In the Search Direction area, specify the direction or angle in which the bar code should be read on the image. For example, select the 0 option to read the bar code from left to right on a document as it is scanned. You can select more than one search direction. Multiple search directions are applied in order from 0 - 270.

8. In the Select Bar Code Types area, highlight a bar code type and select either the **Add to Scan ID** button to add the bar code type to the Scan Identification group or the **Add to Batch Processing** button to add the bar code type to the Batch Processing group. You can add as many bar code types as you want to either group. Select from the following bar code types:

    •   Codabar - Binary bar code of variable length. The valid character set includes 0–9 and the special characters: / . +. Codabar uses series of wide and narrow elements for character representation and does not support the check sum feature.

- Code 128 - Proportional bar code of variable length. The valid character set includes all 128 ASCII characters. This dense bar code is comprised of six elements (three bars and three spaces). With the exception of the stop character, all other characters are 11 units long. Check sum is mandatory with the Code 128 bar code.

- Code 39 (3 of 9) - Binary bar code of variable length. The valid character set includes A–Z, 0–9, and the special characters $ - : / . +. The asterisk (*) is used as the start and stop character. Check sum is optional with the Code 39 bar code.

- Code 93 - Proportional bar code of variable length. The valid character set includes A–Z, 0–9, the space, four shift characters, and the special characters $ - % . + /. This structurally complex bar code requires two check digits. Code 93 does not support the check sum feature.

- EAN (European Article Numbering) - Fixed length code with either eight or 13 characters. The last digit is a check digit. Check sum is mandatory with the EAN bar code.

- Interleaved 2 of 5 - Variable-length bar code that uses five black bars and five white bars to define a number. Check sum is optional with this bar code type.

- Linear 2 of 5 - Variable-length, numeric bar code that uses five black bars to define a digit. Check sum is optional with this bar code type.

- PostNet (Postal Numeric Encoding Techniques) - Numeric-only, binary bar code. The length can vary, but it must be five, nine, or 11 digits. All variations contain one check digit.

- UPC-A (Universal Product Code-A) - Numeric-only, fixed length 12-character bar code designed to identify a product and its manufacturer. This is a standard bar code scheme for items available for sale to the general public. Supplementals are supported.

- UPC-E - Numeric-only, zero-suppressed version of UPC-A. This version compresses the data characters and the check sum into six characters.

9. Define the settings for each bar code type you added to the Scan Identification group. The Scan Identification process identifies forms during scanning. This feature will match your settings against bar codes on hard copy documents to identify each document's form type. To define the scan ID for a bar code type, select the tab for that type and do the following:

   a. In the **Bar Code Data** field, type the data to be matched against the bar code, which is then used to identify the form. You can use an asterisk (*) as a wild character if the values you type do not match the bar code data exactly.

   b. In the **Order Number** field, type the order position of this bar code on the document, from top to bottom and left to right, if there is more than one bar code on the document being scanned.

   c. In the **ID Number** field, type the number associated with all scanned documents that contain this bar code. This number is placed before the corresponding image name in the BDF and is used for Form ID.

10. Define the settings for each bar code you added to the Batch Processing group. In Batch Processing, you define the bar codes that will be used to create batch separators. When the scanning process detects the specified bar code, it generates a new BDF. The bar code type displays in the **Type** field. Select the **Use As Bar Code Segment** check box to assign a bar code as the batch name during the scanning process. This allows you to easily identify a batch based on its bar code. When you select this option, a new string segment is defined that recognizes bar codes. This segment requires the string name generator to use a bar code as the value of the segment.

11. To remove a bar code type from the Scan ID process, select the numbered tab for the bar code type and select the **Remove from Scan ID** button. To remove a bar code type from the Batch Processing group, select the numbered tab for the bar code type and select the **Remove from Batch** button.

12. Select **Save** on the Scanner Configuration Designer window toolbar to save your changes.

Defining Patch Code Scan Settings

You typically use patch codes as a document separator, batch separator, or form/document identifier. Unlike bar codes, patch codes do not contain data.

The Patch Code options are available regardless of whether your scanning hardware supports the patch code functionality. Check your scanning hardware documentation to determine whether your hardware supports patch codes. You can also enable patch code recognition from an Adrenaline card if your scanner does not support patch code recognition. Refer to the documentation for your Adrenaline card for more information. This feature is not available if you are using an Adaptec card in conjunction with the Adrenaline Capture Engine (ACE).

The left side of the Patch Code tab lists the patch code types, and the right side contains tabs, each representing a single patch code. You can specify a number of patch codes to be associated with the image string; however, you will use only one patch code per image string for Form ID during the scanning process.

To set patch code data properties:

1. From Administration, select the **Scan Designer** button. The Scanner Configuration Designer window displays.

2. Select the Data icon on the Config panel.

3. Select the Patch Code tab.
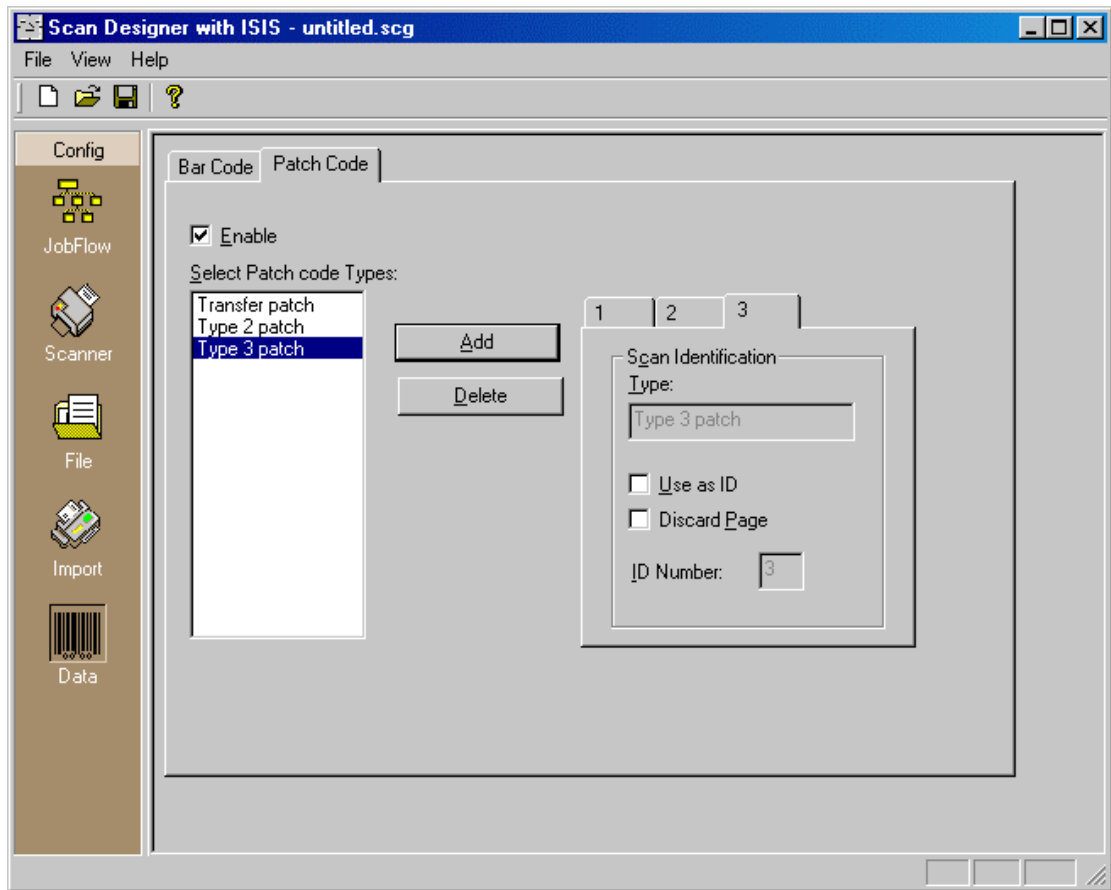
4. Select the **Enable** check box to activate patch code functionality.

5. In the **Offset to Center** field, select the value that indicates the vertical distance from the center of the document to the patch code in inches.

6. In the **Select Patch code Types** field, highlight a patch code type and select the **Add** button. The patch code type determines the kind of patch code on the image. You can add as many patch code types as you want. Following are examples of the patch code types:

7. Select the **Use Scanner If Possible** check box to instruct FormWare to use the scanner hardware to recognize patch codes if the scanner hardware is able to do so. However, your scanner hardware may not recognize all patch code types. Check your scanner hardware documentation regarding the patch code types your scanner hardware recognizes.

8. Select the **Discard Page** check box to instruct FormWare to discard pages that do not have the correct patch code to use in Form ID.

9. In the Batch Processing area, you define the processing settings for the patch code type displayed in the **Type** field. You can set the following options:

   a. Select the **Use as ID** check box to assign a patch code as the batch name during the scanning process. This allows you to easily identify a batch based on its patch code. When you select this option, a new string segment is defined that recognizes patch codes. This segment requires the string name generator to use a patch code as the value of the segment.

   b. In the **ID Number** field, type the number associated with all scanned documents containing the specified patch code. This number is placed before the corresponding image name in the BDF and is used for Form ID.

10. To delete a patch code type, select the numbered tab for that patch code in the Batch Processing area and select the **Delete** button.

11. Select **Save** on the Scanner Configuration Designer window toolbar to save your changes.

Using the Administration Module

The Administration module is the primary entry point into the FormWare system. This module enables you to create and manage an operator-assisted, high-volume, forms processing application. When you create a forms processing application, files and other components are generated that run the application, including Batch Definition files (BDF), batch data files, Job Definition files (JDF), Form Definition files (FDF), and image files (TIFF). You will use the Administration window to manage these files and execute processes over your forms processing application. You can also use the options on the Administration window to create check-digit algorithms, keyboard maps, and other customized pieces necessary for forms processing.

Toolbars and drop-down-menu options enable you to manage you FormWare files. For example, you can select the **Cut**, **Copy**, and **Paste** toolbar buttons to perform these functions on the files in the current folder. You can also conduct ad hoc processing functions such as exporting BDFs and XML files.

With any forms processing system, additional supporting files must be created—not just batches. Items such as value tables and check digits are necessary to validate data and to ensure data integrity. Also, each batch contains a BDF and the images associated with the batch. The Administration module provides the means to manage these necessary folders and files. You can display, delete, and create images, batches, and supporting files used by FormWare.

When you start the Administration module, the Administration window displays the contents of the FormWare system in a hierarchy of folders and files. The tree-style browser allows you to view the contents of each folder by file type and create new files.
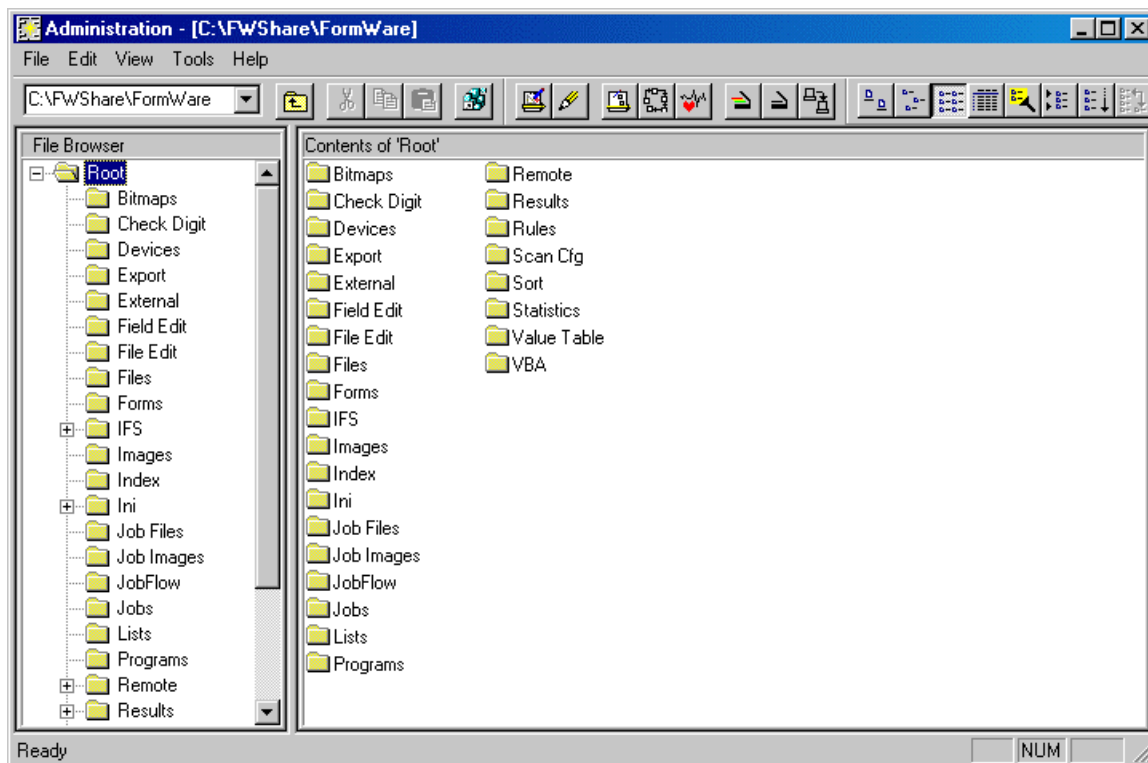
Understanding the File Browser

The **Root Directory** field above the File Browser displays the full path to the root directory where FormWare system folders and files are located. The File Browser displays the directory structure of the root directory. If you have access to other FormWare root directories (e.g., on a network system), you can select another root directory from this field.

The File Browser displays a hierarchy of folders based on the FormWare root directory. When you select a folder, the files contained in the selected folder display in the Contents area. Select the + or – sign next to a level in the File Browser to expand or collapse the subordinate levels. You can also create and navigate subfolders within the browser. You will see the following folders in the File Browser:

- **Bitmaps** - Contains the jpeg (JPG) image files, which are used for bitmap fields in Completion. You can also use bitmaps as form backgrounds or as icons on the form.
- **Check Digit** - Contains the check digits used for checking data validity. You can create up to 64 check-digit algorithms.
- **Devices** - Contains the FormWare components used to import and export data. Using devices, you can create a file repeatedly without specifying the path and file name.
- **Export** - Contains the export programs used to format batch data for printing or for export to a text file, PDF, or other backend repository.
- **External** - Contains the external program source code files. You can compile these code files into Field Edit or File Edit programs.
- **Field Edit** - Contains the Field Edit programs execute during Completion. These programs can contain validation routines, table lookups, or check digit algorithms.
- **File Edit** - Contains the File Edit programs that execute when a batch has finished a Completion session or as a custom JobFlow task.

- **Files** - Contains the batch data files and Batch Definition files (BDF).
- **Forms** - Contains the Form Definition files (FDF) and form help files used during Completion. You create FDFs in Designer by placing fields on data forms.
- **IFS** - Contains the files for the Internet FormWare Server (IFS) . This folder has the following subfolders: Data, Logs, and Scripts.
- **Images** - Contains the images (TIFF or JPG) and recognition results files (OCR).
- **Index** - Contains the indexes used to locate data within an associated batch. Indexing the batch using a specified key creates an index.
- **INI** - Contains the configuration files (INI), which are ASCII text files with parameters that control various parts of a forms processing application.
- **Job Files** - Displays the BDFs associated with a job for maintenance purposes.
- **Job Images** - Contains the image files for each job, facilitating managing the related files contained in different directories.
- **JobFlow** - Contains the JobFlow Definition files (JFD) and queue files used by a JobFlow system.
- **Jobs** - Contains the Job Definition files (JDF) and job help files. A JDF is a collection of FDFs created in Designer.
- **Lists** - Contains dictionary files (LST), which are ASCII files that contain a list of values that can be returned by the recognition engine.
- **Programs** - Contains the executable files used by FormWare.
- **Remote** - Contains subdirectories used by I-Forms and the Distributed Processing components.
- **Results** - Contains output data files from several export modules.
- **Rules** - Contains the rule files (RUL) used to validate and interrogate recognition results and pre-Completion and post Completion batch data. The InScript FDF is also stored in this folder.
- **Scan Cfg** - Contains the Scan Configuration files (SCG) created in the Scan Designer module.
- **Sort** - Contains the sort programs written in the FormWare Programming Language (FPL) and executed against a batch to physically sort the records in the batch.
- **Statistics** - Contains the operator statistics files collected during Completion. You can generate reports from these files.
- **Value Table** - Contains all value table assignments. A value table is a batch upon which you perform table lookups or an index that points to a batch. You can create up to 100 value tables.
- **VBA** - Contains the VBA project files that you create.

Many image-based, forms-processing systems segregate images and recognition results into subdirectories that are usually based on a customer name or Julian date. This means that a job can have associated images and data in many different locations on the system. The Administration module supports virtual directories, enabling you to select one folder, such as the \FormWare\Job Files folder, in which to display all data files in all directories for a specific job. Virtual directories facilitate management of files related to your forms-processing application.

Understanding the Contents Area

The Contents area of the Administration window displays the folders or files for the highlighted folder in the File Browser. For example, if you select the Forms folder in the File Browser, you can view the Form Definition files (FDFs), setup images, and help files related to data forms in the system. To view all files within a folder, select the **Show All Files** toolbar button.

When you double-click on a file in the Contents area, it displays in the relevant editor or module. For example, if you double-click on a TIFF, the TIFF image displays in the Image Viewer. If you double-click on an FDF, the Designer module opens with only that data form loaded.

The root directory indicates the base directory where FormWare folders and files are located. A high-volume forms-processing system might have more than one root directory for FormWare. Before you can access a root directory from the FormWare Root Directory field (other than the one that is installed on your workstation), you must install the FormWare software to that location. Once FormWare is installed, use Administration to access and manage the root directories.

Defining Root Directories

To access a valid root directory, you must add it to the list of available root directories in the Root Directory field that displays above the File Browser.

Adding a Root Directory

To add a new root directory to the list of available root directories:

1. Open the FormWare Administration window.

2. From the Edit menu, select the **Edit Root Directories** option. The Edit Root Directories dialog displays.
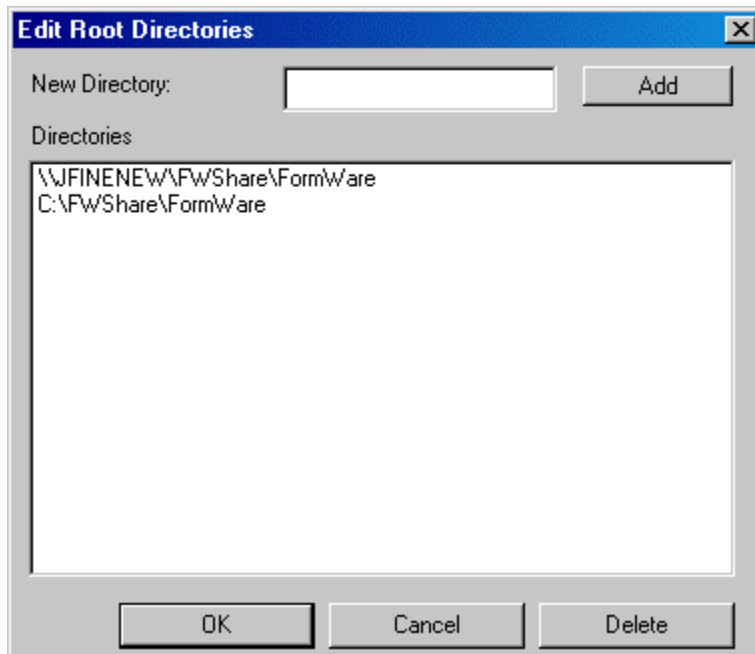
3. In the **New Directory** field, type the path to the valid FormWare directory. You must type a valid path and directory name or you will receive an error.

4. Select the **Add** button to add the root directory to the Directories list.

5. Repeat these steps for each new root directory you want to add.

6. Select **OK** to return to the Administration window.

Selecting a Root Directory

Only one root directory at a time can display in the File Browser of the Administration window. To select a different root directory:

1. From the Administration window, display the drop-down list for the **Root Directory** field above the File Browser.

2. Select a root directory from the list. The File Browser now displays the folder structure for that root directory.

Deleting a Root Directory

To delete a root directory from the list of available root directories in the Root Directory field:

1. From the Edit menu on the Administration window, select the **Edit Root Directories** option. The Edit Root Directories dialog displays.

2. Highlight the directory you want to remove in the Directories list and select **Delete**. This removes the directory from the list of root directories, but does not delete the directory. Repeat this step for each root directory you want to delete.

3. Select **OK** to return to the Administration window.

Selecting a New Directory for Files

If you want to change the default directory for files, do the following:

1. Open Administration.

2. In the File Browser, open the Files folder.

3. From the Edit menu, select the **Current Directory** option. The Select Directory dialog displays.

4. Select the **Previous Folder** button to navigate to another directory and select **OK**.

Limiting Files in the Contents Area

You can limit the files displayed in the Contents area of the Administration window based on a filename pattern, range of filenames, or the file time or date. To do this:

1. Open Administration.

2. Select the **Limit File Selection** button on the File Selection toolbar. The Select Files dialog displays.

3. To limit files based on a file name pattern, type a file name pattern in the **Pattern** field on the **Pattern** tab. The Contents area will display only files that contain this pattern. You can use an asterisk as a wildcard in the pattern. Select the **Mark** check box to highlight all files with the specified pattern in the Contents area.
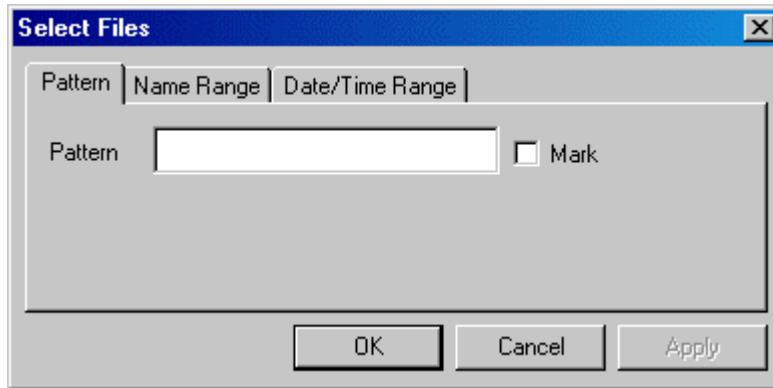
4. To limit files based on a range of file names, select the **Name Range** tab.
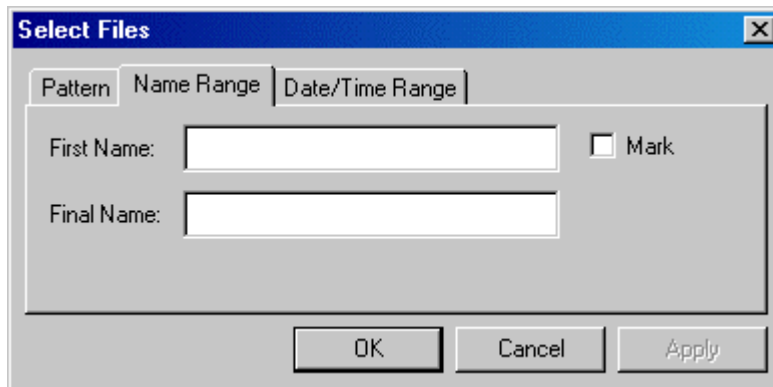
5. Enter the following values on the **Name Range** tab:

   a. In the **First Name** field, type the first filename in the range of filenames to select.

   b. In the **Final Name** field, type the last filename in the range of filenames to select. All files with names between and including the First Name and the Final Name display in the Contents area.

   c. Select the **Mark** check box to highlight all files within this name range in the Contents area.

6. To limit files based on the file date or time, select the **Date/Time Range** tab.
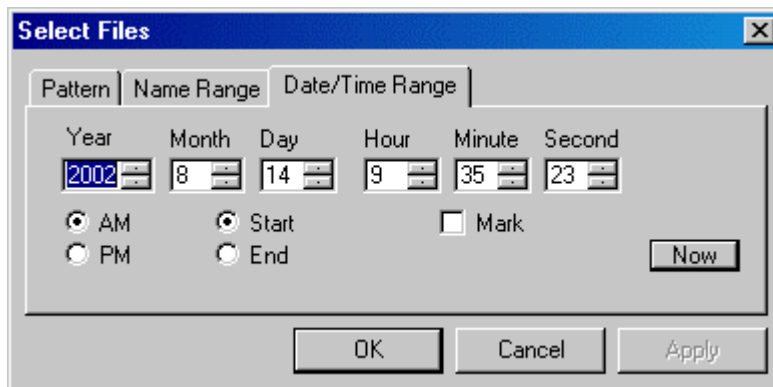
7. Enter the following values on the **Date/Time Range** tab:

   a. In the **Year/Month/Day/Hour/Minute/Second** fields, select one or more date and time values to limit the Contents area to files within this date/time range. Use only those fields that you want to apply to the date/time range selection.

b. In the **AM/PM** fields, select the value that applies to the **Hour** field.

c. Select the **Start** or **End** check box to define whether the date/time values are the beginning or the end of the search range. Use these fields together to define both the beginning and the end of the date range. For example, select the **Start** field and type values in the date and time fields to establish the start of the search range. Then select the **End** field and type date and time values for the end of the search range.

d. Select the **Now** check box to insert the current date and time into the date and time fields.

e. Select the **Mark** check box to highlight all files that fall within the search parameters in the Contents area.

8. Select **Apply** on the Select Files dialog to display the files based on the criteria set and keep the dialog open. Select **OK** to apply the criteria and close the dialog.

Running Programs from Administration

You can launch all FormWare stand-alone programs from the Administration window. This lets you scan or import new images, start a JobFlow Client, or develop a forms-processing application using the Designer module. You can start components from the Administration window in the following ways:

- Select the component from the Tools menu.
- From the Accessories toolbar, select the button for the component.
- In the File Browser, open the Programs folder in the File Browser and double-click on the component program file in the Contents area.

When you exit a FormWare module, the program closes and you return to the Administration window if you launched the program from Administration.

Using the Image Viewer

The Images folder is the default directory for forms processing images. It contains all of the images and recognition results files used in the FormWare root directory. Images used by the FormWare system can be Group3 or Group4 TIFF, as well as JPEG file formats. Images can have .TIF or .JPG file extensions. The recognition results files have an .OCR extension. OCR files contain the data resulting from recognition performed on the images in a batch.

From the Images folder you can use the Image Viewer to look at sections of any image in detail or see a thumbnail of the entire image. The black area of the thumbnail corresponds to the enlarged area of the image displayed in the viewer.

Note: If the image is not visible (gray thumbnail), your color palette settings are incorrect. The thumbnail functions only with a setting of 256 colors – not 16 colors, high color, or true color settings.

To view an image:

1. Open the Images folder in the Administration File Browser.

2. Double-click on any image file in the Contents area. The image displays in the Image Viewer dialog. The name of the image you selected displays as part of the header name. Here is an example of the Image Viewer dialog with the Image Thumbnail viewer:
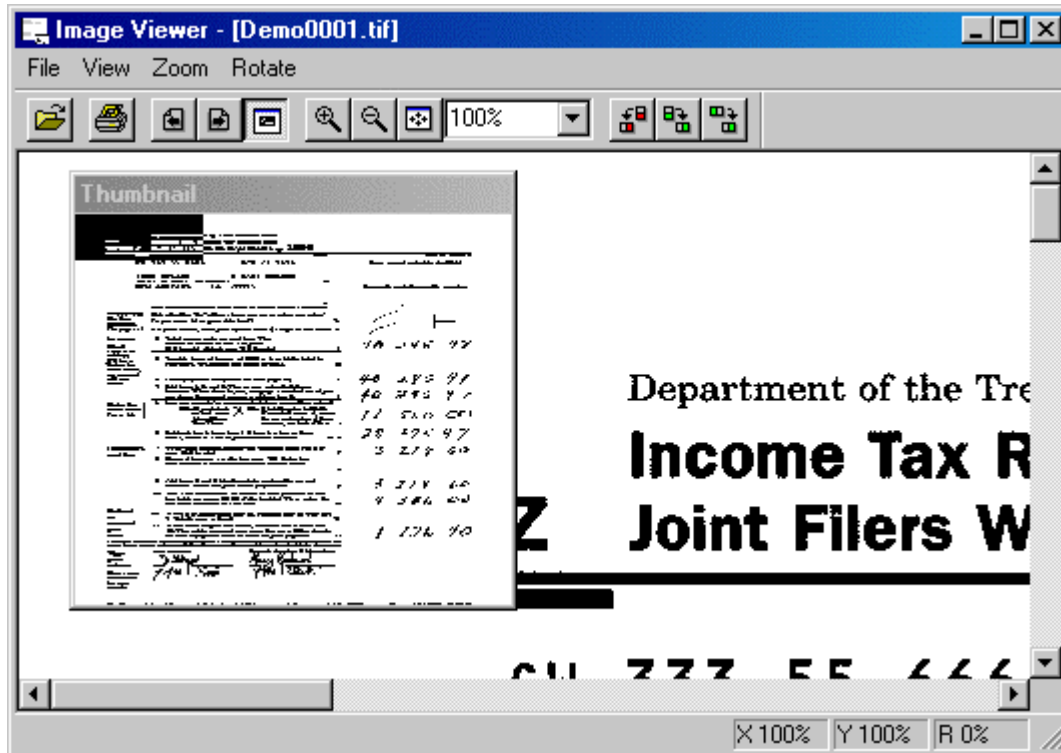
3. Use the Image Viewer Tools toolbar to perform the following functions on the image within the dialog:

    o **Open Image -** Displays the Open dialog where you can select another TIFF image to open in the Image Viewer. The file type on this dialog defaults to TIFF files, but you can change it to display all files within the Images folder.

    o **Print Image -** Displays the Print dialog where you can set options to print the image file.

    o **Previous Image -** Displays the previous image in the series. This option is enabled when images have sequential filenames (e.g., demo001.tif, demo002.tif).

    o **Next Image -** Displays the next image in the series. This option is enabled when images have sequential filenames.

    o **Thumbnail -** Displays a miniature view of the image in the Thumbnail dialog, which is superimposed over the Image Viewer dialog. Select this button again to close the Thumbnail dialog. When displayed, the black portion appearing on the Thumbnail image represents the portion of the image visible in the Image Viewer. Use the mouse to drag the black box and change the position of the image in the Image Viewer dialog.

    o **Zoom In -** Enlarges the image in the Image Viewer in ten percent increments.

    o **Zoom Out -** Shrinks the image in the Image Viewer in ten percent decrements.

    o **Scale To Window -** Resizes the entire image to fit in the current Image Viewer dialog. **Scale To** toggles with **Zoom**.

    o **Zoom -** From the drop-down arrow on this field, select a percentage zoom for the image. Percentages are in increments of ten percent. The default value for all images is 100%. **Zoom** toggles with **Scale To Window**.

    o **Rotate Left -** Rotates the image counterclockwise in 90º turns.

    o **Rotate Right -** Rotates the image clockwise in 90º turns.

o **Flip -** Flips the image (i.e., rotates it 180º).

4. To exit the Image Viewer dialog, select **Exit** from the File menu.

Working with Batches

Batches are controlled and managed by the following types of files:

- **Batch data files**, which contain all of the data generated by the recognition engine and the reject repair operator. These files are created in the \FormWare\Files folder without an extension after you have run a batch through the recognition process.

- **Batch Definition files (BDF)**, which contain the information about a batch (e.g., the images in the batch) as it processes through the system. These files have a .bdf extension.

- **Batch log files**, which contain audit information about the batch. You can identify batch log files in the Files folder by their .log extension.

Creating New Batch Files

To create a new batch data file, BDF, or batch log file:

1. Select the Files folder in the Administration File Browser.

2. Right-click on the Contents area and select the **New** submenu from the batch file context menu. Then, select one of the following options:

    o **Batch** - Creates a new batch, launches the Completion module, and starts the batch. Remember that image processing requires an associated BDF for the new batch.

    o **Batch Definition File** - Creates a new BDF with a .bdf extension and launches a text editor so you can type the image names. The BDF contains a list of all images for the batch and parameters for job name, image directory, and recognition results file directory.

    o **Batch Log** - Creates a new batch log file with a .log extension and launches a text editor for creating a new log file. The log file has no defined structure. You can enter any desired text in the file.

Note: All BDFs should end with a blank line as the last line in the file (i.e., a single carriage return). If you do not insert this extra line at the end of the BDF file you may get unexpected results at process time.

Working with Batch Data Files

You distinguish batch data files from other file types in the Files folder in that they have no file extension. Within the Administration Files folder, you can perform the following tasks on batch data files:

1. Open the Files folder in the Administration File Browser to display the folder contents. If files are not visible in this folder, select the **Show All Files** button on the toolbar.

2. Double-click on a single batch data file to launch the Completion module and displays the batch in Correct mode.

Or,
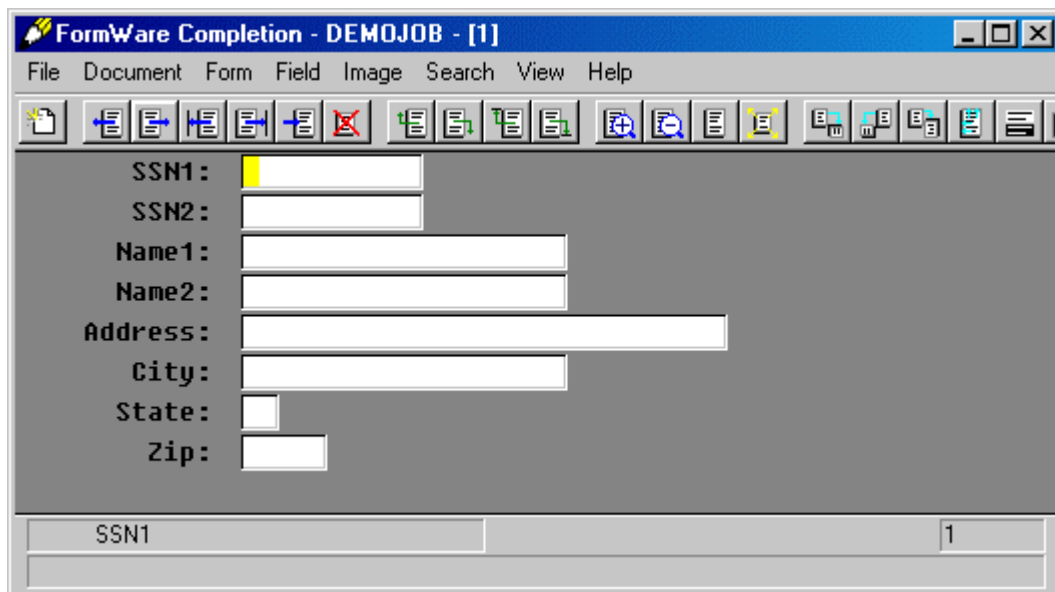
Highlight one or more batch data files in the Contents area. Use **[Ctrl-Click]** to select multiple files. Right-click to display the Contents menu. You have the following options on the Contents menu for batch data files:

- **Cut** - Cut the selected file(s) and store in temporary clipboard memory.
- **Copy** - Copy the selected file(s) and store in temporary clipboard memory.
- **Paste** - Paste the file(s) stored in temporary clipboard memory to current folder location.
- **Delete** - Delete the selected file(s).
- **Rename** - Rename the selected file. Only works on one file at a time.
- **Properties** - Display the general files properties for the selected file. Only works on one file at a time.
- **Send To > Batch Data Viewer** - Export the contents of the batch data file to a text editor. This creates a temporary text file, which is stored in the \FormWare\Tmp folder.
- **Send To > Hex Utility** - View the contents of the selected batch data file in hex format. Press [Enter] at the command line prompt to display the hex information for the selected file. Only works on one file at a time.
- **Send To > File Recovery** - Recovers a damaged or corrupt batch data file using the FormWare recovery function. Only works on one file at a time.
- **Edit** - View the contents of the batch data file in the Completion module and display the batch in Correct mode. On batch data files, the Edit function works the same as double-clicking a batch data file in the Contents area.

Note: If an Contents menu option is not listed here, then it does not apply to the selected file type. To create a new file within this folder, refer to the Creating New Batch Files task.

Working with Batch Definition Files

FormWare requires that a Batch Definition file (BDF) be created when image processing is performed. BDFs can exist anywhere in the FormWare root directory, although the

\FormWare\Files directory is the default location. A BDF is a text file with the following basic structure:

```
JOBNAME=HCFA1500
F:\FORMWARE\IMAGES\TEST0001.TIF
F:\FORMWARE\IMAGES\TEST0002.TIF
F:\FORMWARE\IMAGES\TEST0003.TIF
```

You distinguish BDFs from other file types in the Files folder by their .BDF extension. Within the Administration Files folder, you can perform the following tasks on BDFs:

1. Open the Files folder in the Administration File Browser to display the folder contents. If files are not visible in this folder, select the **Show All Files** button on the toolbar.

Note: To locate BDFs in a folder other than the \FormWare\Files folder, open the Files folder in the File Browser and select the **Current Directory** option from the Edit menu. The Select Directory dialog displays, which allows you to move up the tree structure to select a new folder from within the FormWare tree or from a folder outside the tree. To return to the Files folder, you must select the Files folder in the File Browser.

---

2. Double-click on a single BDF to launches the text editor where you can view, edit, and save changes to the BDF.

Or,
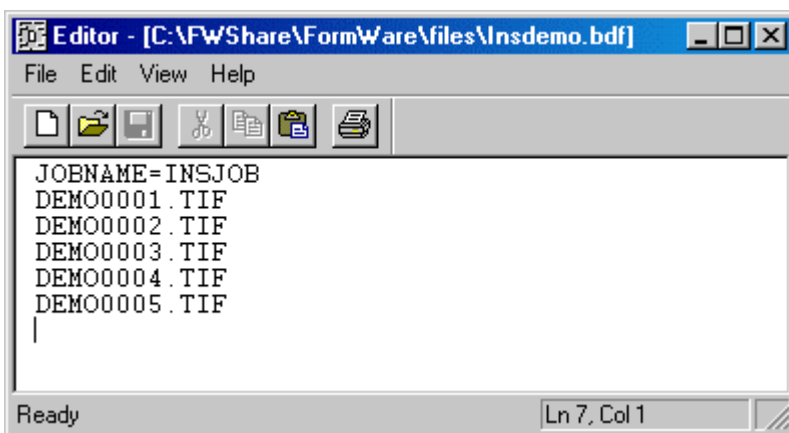
Highlight one or more BDFs in the Contents area. Press **[Ctrl]-[Click]** to select multiple files. Right-click to display the Contents menu. You have the following options on the Contents menu for BDFs:

*   **Cut** - Cut the highlighted file(s) and store in temporary clipboard memory
*   **Copy** - Copy the highlighted file(s) and store in temporary clipboard memory
*   **Paste** - Paste the file(s) stored in temporary clipboard memory to current folder location
*   **Delete** - Delete the highlighted file(s)
*   **Rename** - Rename the highlighted file. Only works on one file at a time.
*   **Properties** - Display the general files properties for the highlighted file. Only works on one file at a time.
*   **Send To > Text Editor** - Displays the contents of the BDF in a text editor. This functions the same as either double-clicking on the BDF file in the Contents area or selecting the Edit option on the Context menu. Only works on one file at a time.

- **Send To > Hex Utility** - View the contents of the highlighted BDF in hex format. Press [Enter] at the command line prompt to display the hex information for the selected file. Only works on one file at a time.

- **Edit** - View the contents of the BDF in a text editor. On BDFs, the Edit function works the same as either double-clicking a BDF in the Contents area or selecting the Send To > Text Editor option on the Context menu. Only works on one file at a time.

- **Print** - Print the contents of the BDF. Displays the Print dialog where you can direct the print job to a printer of your choice.

Note: All comments should be added to a BDF as properties where the name of the comment property is user-defined (e.g., COMMENT=This is a comment.). For each comment line in the BDF, you must use a unique property (e.g., COMMENT1=, COMMENT2=).

Accessing the Designer Module

You will use the Designer module to create and define all the elements of a job, including the forms and fields for your application. To start the Designer module:

1. From FormWare Administration, select the **Designer** button on the toolbar. The Designer module window displays.

2. To open an existing FormWare application within Designer, select the **Open** button on the toolbar and browse for the job you want to modify. Jobs, identified by their .jdf extension, are located in the \FormWare\Jobs folder. Select the job and the **Open** button to display the job in the Designer window.

   Or,

   To create a new FormWare application, select the **New** button on the toolbar. See the steps in the Creating a Job section for more information.
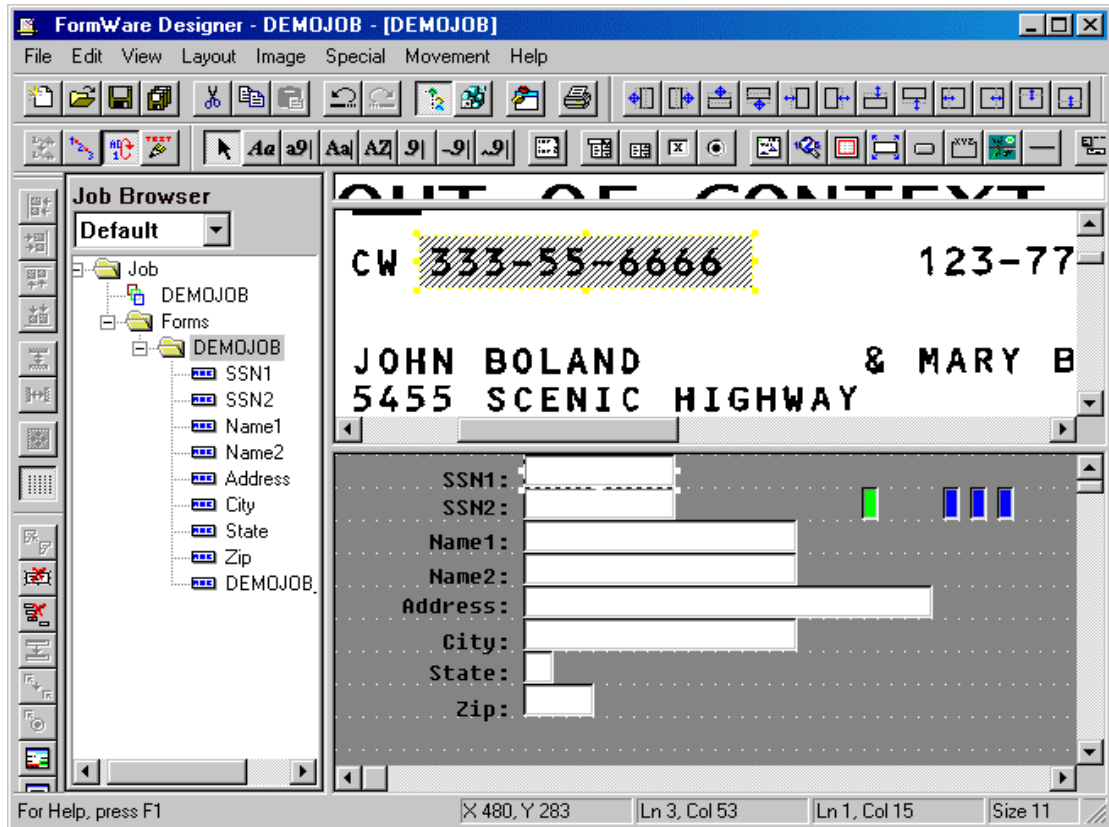
The Designer window contains the following user interface components:

o   **Menu Bar -** The following menus are available on the Designer window: File, Edit, View, Layout, Image, Special, Movement, and Help. The options on these menus enable you to perform functions as you design your FormWare application. Many of the features you access on these menus correspond to buttons on the toolbars.

o   **Toolbars** - The following toolbars are available on the Designer window: Standard, Layout, Special, Image, Zone Movement, and Field Types. The buttons on these toolbars enable you to perform functions as you design your FormWare application. Many of the features you access on the toolbars correspond to options on the menus.

o   **Job Browser** - View and navigate the objects in the job and data forms on the left side of the window. The Job Browser displays an icon for the job, the data forms in the job, and the data fields on each data form. You can use the field at the top of the Job Browser to sort the fields within the Job Browser view. There are three field sort options: Default (sorted in order of field creation); Ascending (sorted alphabetically, A to Z); and Descending (sorted alphabetically, Z to A).

o   **Reject Ticker Tape** - Displays the number of lines of out-of-context characters visible during a Completion reject repair session at the top of the window. You can stretch the bottom of this area to increase the number of lines that display or extend it to the top of the window to eliminate the display of out-of-context characters. To close the reject ticker tape, select the bottom border of this area with your mouse and drag it to the top of the main Designer window.

o   **Image Viewport** - Determines the portion of the image presented to the operator during a Completion session. In the Designer module, the setup image displays in the Image Viewport. To increase or decrease the amount of the image displayed during Completion, you can stretch the bottom of this window by selecting the top or bottom

bottom of the Designer window. A data form is a collection of fields that define the data you want to extract from an image or paper form. The data form typically replicates the arrangement of fields and labels on the paper form. This is the location where you will place the fields and labels for each form in your FormWare application.

o   **Status Bar** - Provides information about the currently selected object. The first section displays the X and Y coordinates of the location of the mouse within the Image Viewport. The second section displays the current location of the mouse in the data form, while the third displays the coordinates of the field based on the Line and Column of the grid. The fourth and fifth sections display the size and location of the field.

3.   Before exiting the Designer module, select the **Save All** button to retain your work.

4.   Select **Exit** from the File menu to close the Designer module and return to FormWare Administration.

Modifying System Default Properties

If you are creating several jobs to which you want to apply system defaults, you would use the System Defaults dialog to set global properties for all your jobs. When you create a new job, the Job Defaults dialog displays so that you can override the system defaults for the new job.

Setting System Default Properties

To set the system default properties:

1.   Within the Designer module, select **System Defaults** from the Edit menu. The System Defaults dialog displays the **Form and Field tab**.
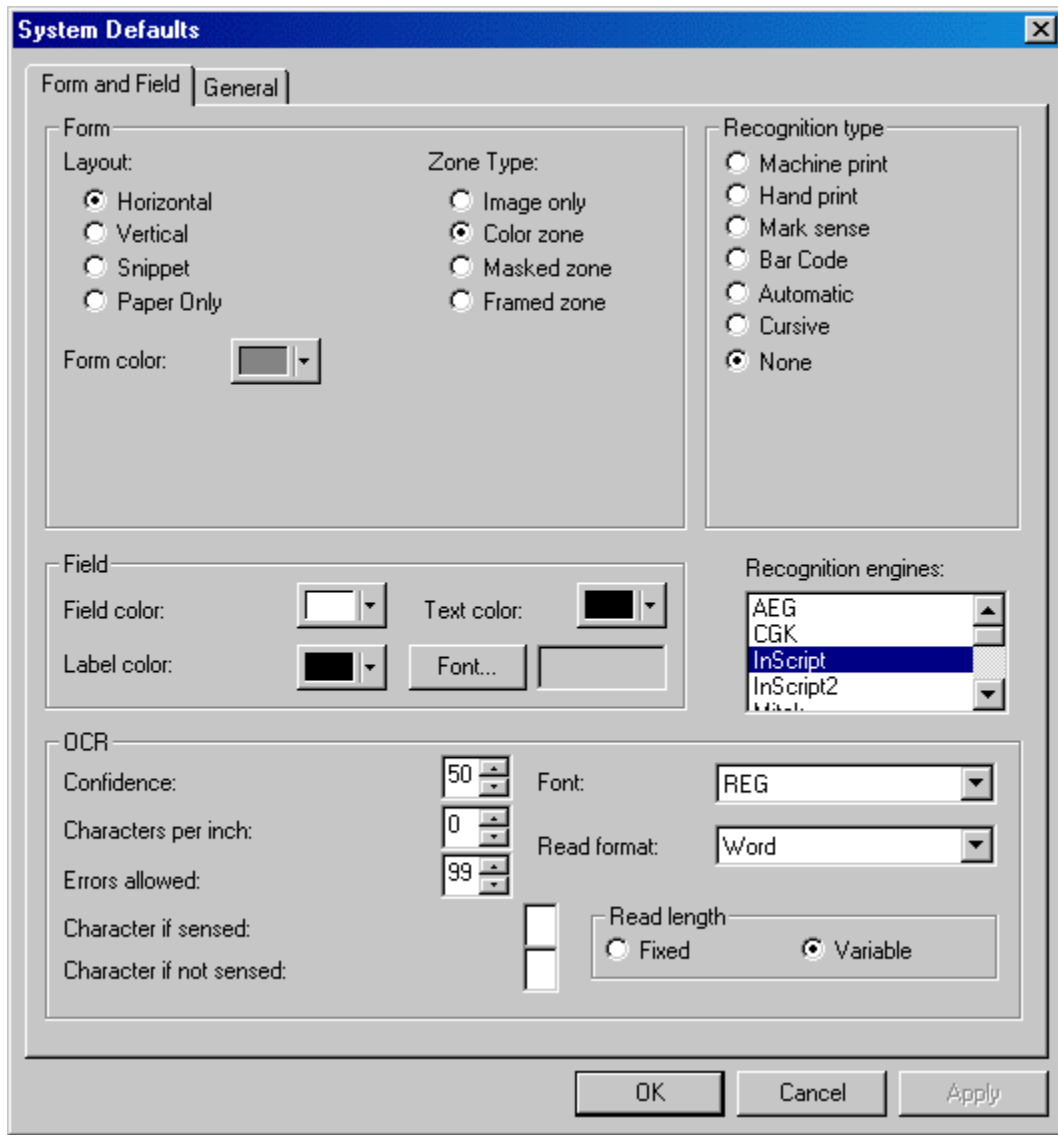
Using the options on the Form and Field tab you set default properties for new jobs and data forms. This is the only way to set properties for more than one data form at a time. Except for the form layout option, you can change any of these properties on a form or field level, once the form or field is created.

2. In the **Layout** area, select the default layout. This option determines how the image and form display in both the Designer and Completion windows. Options are:

   o **Horizontal** – Displays the image at the top and the data form on the bottom of the window. This is the default.

   o **Vertical** – Displays the image on the left and the data form on the right of the window.

   o **Snippet** – Displays pieces of the image (snippets) interspersed with the fields on the form.

   o **Paper Only** – Indicates a paper-only application. No image displays.

3. In the **Zone Type** area, select the option that determines how you want zones to display on the image. Options are:

   o **Image only** – Displays the image in the Viewport (cannot be used with Recognition).

- o **Color zone** – Displays the image in the Viewport and highlights the recognition zone(s). This is the default value.
- o **Masked zone** – Masks the image and displays only the recognition zone(s).
- o **Framed zone** – Displays the image and frames the recognition zone with a thin red line.

4. In the **Form Color** field, select the arrow to display the Color window and select a default color for the form background. Select a default color or select the **Define Custom Colors** button and use the color palette to define a new color. Select **OK** to save your default color.

5. In the **Recognition type** area, select a default recognition type for fields. Options are:
   - o **Machine print** – Recognizes machine-printed alphanumeric characters that have consistent, predictable shapes.
   - o **Hand print** – Recognizes alphanumeric characters that vary in shape, such as hand-printed text characters.
   - o **Mark Sense** – Recognizes boxes that are marked or checked.
   - o **Bar Code** – Recognizes bar code data.
   - o **Automatic** – Determines the type of characters to be recognized.
   - o **Cursive** – Recognizes cursive handwritten characters.
   - o **None** – No recognition will be performed.

6. In the **Field** area set the default color and font for labels, fields, and text within this job in the following fields:
   - o **Field color -** Displays a color palette, from which you select a color for the field background.
   - o **Text color** - Displays a color palette, from which you select a color for the field text.
   - o **Label color** - Displays a color palette, from which you select a color for the label text.
   - o **Font** - Displays the Font dialog, from which you select the default font, font style, point size, and other characteristics for fields you create on the form. You can override this setting for individual fields.

7. In the **Recognition engines** area, select a recognition engine to use as the default. This engine is applied to all the fields created in the data form. Select from the following engines:
   - o **AEG** - Effective on machine print recognition, including special fonts; dropout mark sense,;out-of-context alphanumeric handprint, including upper and lower case alpha and numeric. The AEG engine also supports a number of foreign languages.
   - o **CGK** - Effective at recognizing contextual alphanumeric handprint, dropout mark sense, and machine print. You can use Trigram analysis and dictionaries with CGK. Unlike the other recognition engines that support field-level dictionaries, CGK creates a form-level dictionary. The CGK engine also supports a number of foreign languages.
   - o **InScript** - Effective at recognizing hand-printed numeric and uppercase alpha, hand and machine print dropout and non-dropout mark sense, and machine print. The InScript engine uses the Caere engine and neural network voting for machine print.
   - o **InScript2** - Effective at recognizing machine print for English and 110 additional languages, machine print for both numeric and alphanumeric characters, handprint for both numeric and alphanumeric (upper and lowercase), OCR-A and OCR-B machine print, MICR (E-13B) machine print, mark sense hand-filled data (unconstrained areas or square-, rectangular-, circle-, or oval-constrained boxes, with

or without dropout boxes), and 1-dimensional (13 symbologies) and 2-dimensional (PDF 417) bar code recognition.

- o **Mitek** - Effective at recognizing dropout mark sense and out-of-context alphanumeric handprint, including upper and lowercase alpha. It also supports CAR and LAR recognition.

- o **Nestor** - Effective at recognizing dropout mark sense and out-of-context alphanumeric handprint, including upper and lowercase alpha. Nestor uses a version of the Caere engine for machine print. It can make effective use of dictionaries to improve recognition results.

- o **OMR** - Effective at recognizing single and multiple dropout and non-dropout mark sense only.

- o **Other** - Use another recognition engine that is not on this list.

- o **Parascript** - Effective at recognizing upper and lowercase handprint and cursive print for fields that have a limited set of possible field values listed in a dictionary, including first names, last names, and addresses. Recognition performance is highly correlated with the quality, completeness, and sparseness of the dictionary.

- o **RecoStar** - Incorporates the AEG and CGK recognition engines and includes an internal voting combination process. RecoStar takes the best of AEG and CGK recognition abilities, and through the use of an advanced engine configuration parameters dialog, enables you to select from voting accuracy bias settings to optimize recognition performance.

8. In the **Confidence** field in the OCR area, select the number that indicates the minimum confidence level for recognized characters.

9. In the **Characters per inch** field, select the number of characters per inch of print to allow on the image.

10. In the **Errors allowed** field, select the number of errors to allow in a field before the field is rejected.

11. In the **Character if sensed** field, select the character to return if the recognition engine determines the mark sense box is filled.

12. In the **Character if not sensed** field, select a character to return if the recognition engine determines the mark sense box is empty.

13. In the **Font** field, select the font pattern to use for OCR. Options are REG, OCRA, OCRB, MICR, or Farrington. For most instances, choose REG.

14. In the **Read Format** field, select the type of format OCR uses to read the data. Options are:

- o **Word** – Spaces are not allowed in the field during recognition. Any space detected is not preserved and the data is compressed.

- o **Line** – Spaces are allowed in the field. If spaces are recognized, they are preserved.

- o **Paragraph** – Enables multiple lines in a single recognition zone.

15. In the **Read length** area, select the option that indicates how OCR will read the length of the field. Options are:

- o **Fixed** – The number of characters in the field during Recognition needs to equal the length of the field. Otherwise, the field is rejected.

- o **Variable** – The number of characters in the field during Recognition does not need to equal the length of the field.

16. Select **Apply** to enable your settings on this tab.

17. Select the **System Defaults, General** tab to set default behavior for maintaining job and job-related files within Designer. These options to determine how the Designer module launches, how property dialogs display, and which files are loaded. The General tab also includes support options for I-Forms.
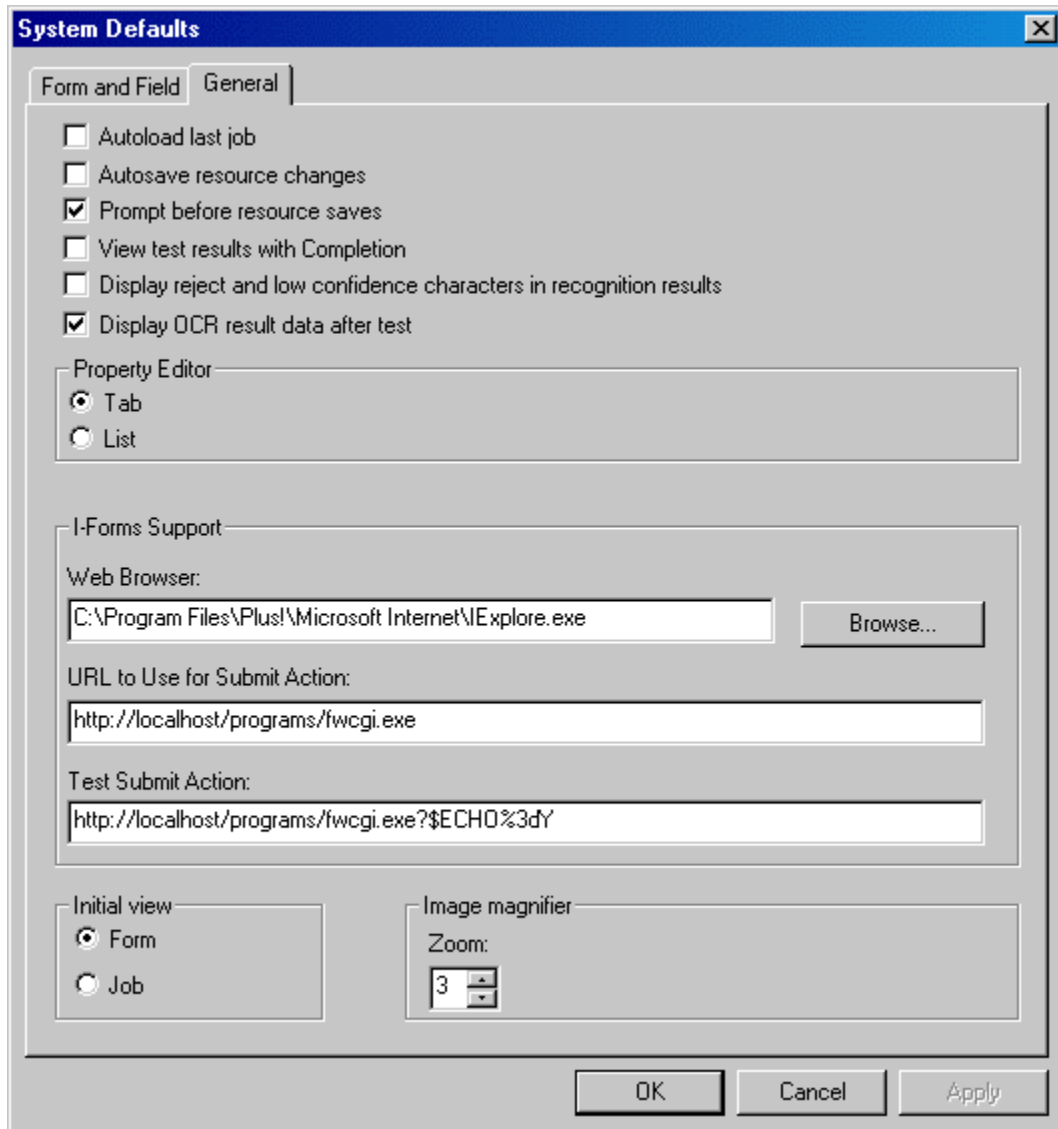
18. Select or deselect the following options on the General tab:

   o **Autoload last job** - Loads the job last opened in Designer when you open the Designer module again. If you do not select this option, Designer opens without a job loaded.

   o **Autosave resource changes** - Saves changes made to any resource (e.g., job, data form, field) without prompting.

   o **Prompt before resource saves** – Save confirmation displays when you close a file or the Designer module. You can choose to accept changes or cancel them from the confirmation dialog.

   o **View test results with Completion** - Displays test results performed within Designer at the field, form, and job level in the Completion window. If you do not select this

option, test results display either in the fields on the Designer window data form or within a component progress dialog.

Note: You can also toggle this option for your application on the fly using the **View Tests in Completion** button available on the Special toolbar of the Designer window.

- o **Display reject and low confidence characters in recognition results** - Instructs the recognition engine to indicate rejected and low confidence characters found during Recognition. When you perform recognition testing in Designer, the rejected and low confidence characters display as question marks in the fields on the data form. You can change the character that displays by using the Error character setting on the Job Properties, Recognition page in Designer.
- o **Display OCR results after data test** - Displays detailed OCR recognition results on the Designer window after a form or field-level recognition test. The corresponding OCR data results for the fields display beneath the Job Browser on the Designer window. The highlighted zone within the Image Viewport is synchronized with the field currently selected on the data form. Each recognized character within the zone that is equal to or above the field confidence level is contained within a colored character clip rectangle. See the Performing a Field or Form Recognition Test in Designer section for more information.

19. In the **Property Editor** area, select the option that determines the default view for all job, form, and field level property dialogs within Designer. Options are:
    - o **Tab** - Displays job, form, and field property dialog pages in a tab view. Each labeled tab represents a page on the property dialog that contains properties related to that label. Select on a specific tab to display the page you want. This is the default view for property dialogs in Designer.
    - o **List** - Displays each of the property dialog pages in a list view. Each item in the list represents a page on the property dialog that contains properties related to that item. Select a specific item in the list to display the page you want.

    See the Changing Property Dialog Views within Designer section for more information.

20. In the **I-Forms Support** area, set the following paths to create default information for any I-Form you create. The URL to Use and Test for Submit Action options can be overridden at the job level on the I-Forms page of the Job Properties dialog.
    - o **Web Browser** - Type the path and program name or select **Browse** to locate for the web browser your PC is running. This enables you to test your forms and view them in the browser, if needed.
    - o **URL to Use for Submit Action** - Type the URL (Universal Resource Locator) for the location on your Web Server to route submitted I-Form transactions for processing.
    - o **Test Submit Action** - Type the URL for the location on your Web Server to which an I-Form is submitted when you test within Designer.

21. In the **Initial view** area, select the level of the Job Browser and properties dialog activated when an existing job opens. Select **Job** or **Form**, which displays the first data form. The default setting is **Form**.

22. In the **Image magnifier** area, change the **Zoom** setting to change the default zoom view for the image. Press [Ctrl+Click] to toggle the image magnifier on and off within the Image Viewport in the Designer window. Press [Ctrl+Right-Click] to scroll between the available zoom magnifications. Valid zoom options range from 2 (i.e., 2 times normal view size) to 12. The default is 3.

---

23. Select **Apply** to retain your settings.

24. Select **OK** to close the System Defaults dialog and return to the Designer window.

Changing Property Dialog Views within Designer

You can quickly change the view of property dialogs within Designer to a view that you find easier to use. There are three levels of properties within the Designer modules: job, form, and field properties. There are two different view types that exist for you to use:

Tab View - Displays each of the property dialog pages in a tab view. Each labeled tab represents a page on the property dialog that contains properties related to that label. Click on a specific tab to display the page you want. This is the default view for all property dialogs within Designer. For example, here is the Job Properties, General page in tab view:
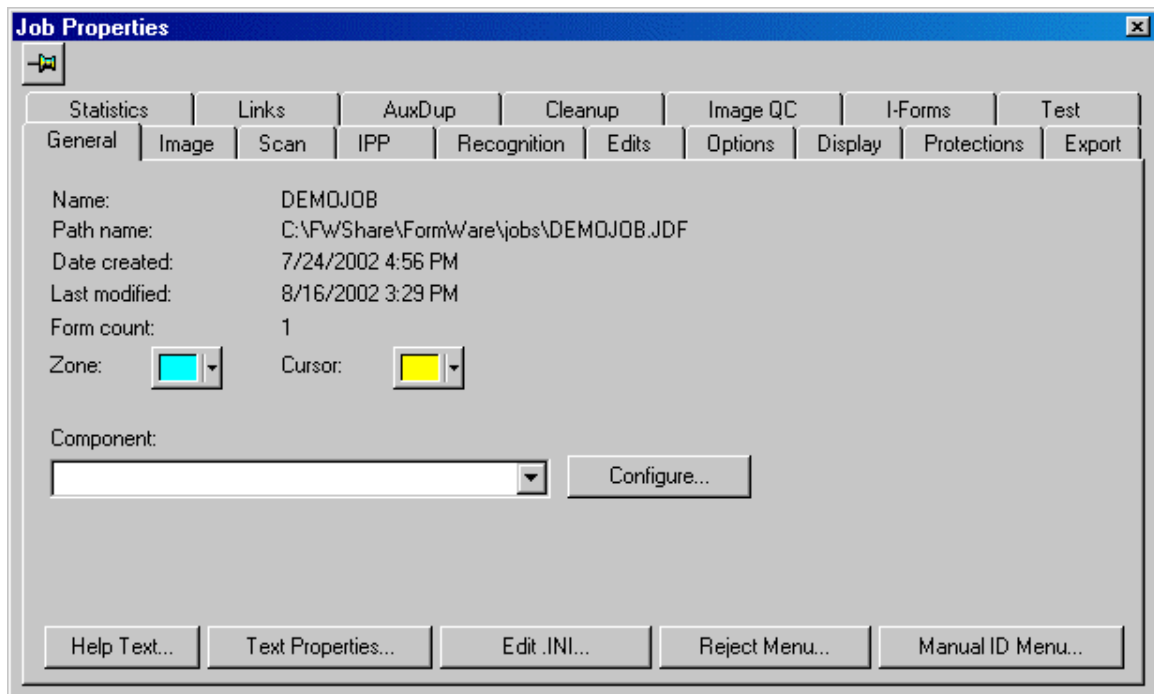
List View - Displays each of the property dialog pages in a list view. Each item in the list represents a page on the property dialog that contains properties related to that item. Select a specific item in the list to display the page you want. For example, here is the Job Properties, General page in list view:
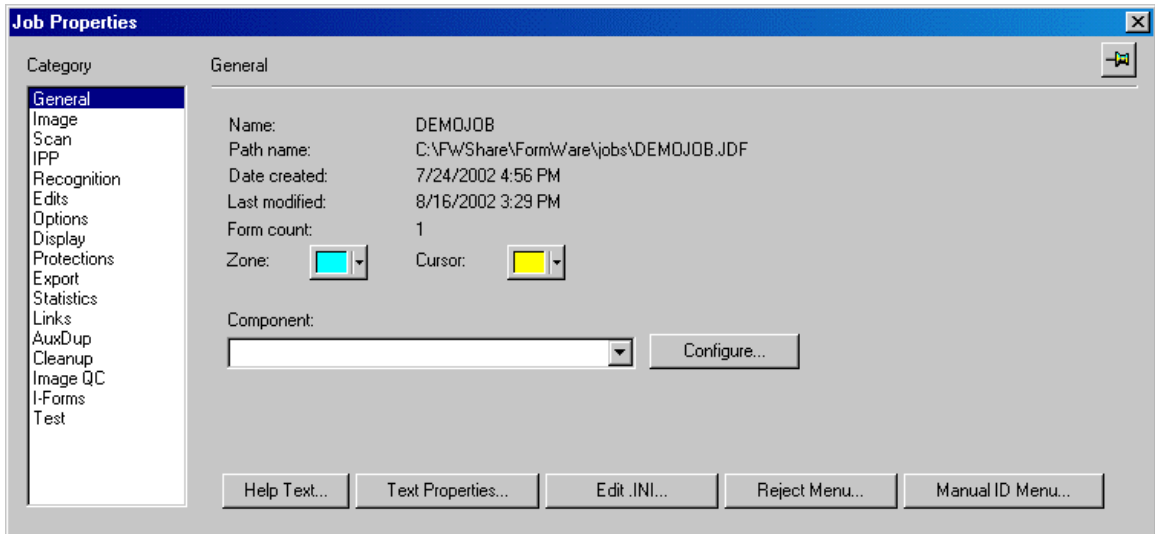
To change property dialog views within the system defaults in Designer:

1. Open a job within Designer.

2. Select **Edit > System Defaults** to display the System Defaults dialog.

3. Display the **System Defaults, General** tab.

4. In the **Property Editor** section, select the appropriate option:
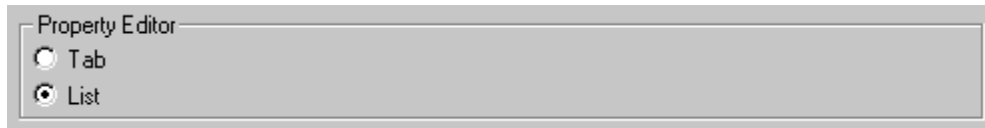
5. Select **Apply** and then **OK** to accept your changes.

To change property dialog views on the fly within Designer:

1. Open a job within Designer.

2. Select **View > Property Editor** and then select the appropriate option.
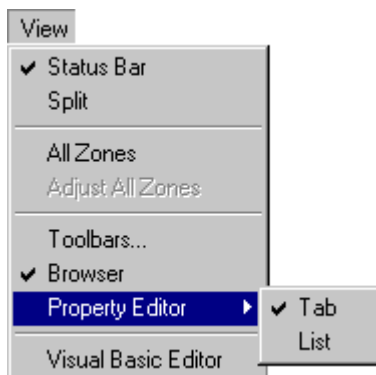
3. Save your changes on the Designer window before you exit.

Note: Changes you make to the property dialog view on either the System Defaults dialog or on the View menu affect the entire system, not just the job. These settings are saved and will be applied to all future jobs, even if you exit the current job and select another.

Setting INI Parameters

INI (initialization or configuration) files are ASCII text files containing functions that control the various parts of your forms-processing application. The functions within INI files are called parameters. Many INI parameters have corresponding options within the user interface, such as in the Designer module. Each parameter must be placed on a line by itself followed by a carriage return; however, blank lines between parameters are prohibited.

See the INI Parameters in the Supplementary Information section for a full list of all INI parameters including syntax and descriptions.

By creating user or group INI files, you can grant limited rights to the maintenance functions of the Administration module and to the various modes of the Completion module. Certain INI file settings control security functions within the FormWare system. The INI files you can access within the Administration control access to the FormWare administration functions and control the users who can access files for the Reject Repair, Verification, or Remove Flags modes of completion.

INI files are located in the \FormWare\INI directory. This directory contains the following subfolders:

- **Fldmap** - The functionality of this folder and its contents are reserved for internal use.
- **Group** - This folder contains INI files that identify groups of FormWare users. A group is a collection of FormWare users to whom you want to assign certain access and user rights.
- **Job** – This folder contains <Job name>.ini files that contain parameters specific to a particular job.
- **JobFlow** - The functionality of this folder is unavailable.
- **Keyboard maps** - This folder contains custom keyboard configurations used by the Completion module.
- **System** - This folder contains information used by all FormWare applications. A typical FormWare installation includes System.ini, Insclnt.ini, Stats.ini and SendTo.ini. Recognition engines, the Completion module, and the Administration module use these files.
- **User** - This folder contains INI files for individual FormWare users. Because each user has his own INI file, each operator or supervisor can have a unique custom environment.

Creating a User INI File

User INI files contain the parameters used during Completion for individual operators. Each file must have the unique name (.ini extension, no periods, and no spaces) with which the operator logs into Windows. To create a user INI file:

1. From the Administration window, open the \FormWare\INI\User folder.

2. Right-click in the Contents area and select **New** from the context menu. A text editor displays.

3. Type the desired parameters in the file. See INI Parameters in the Supplementary Information section for a list of the available parameters you can use within User.ini files.

4. Select **Save** on the toolbar. The Save As dialog displays.

5. In the **File name** field, type the name of the user. This name must match the user's name on the system.

6. Select **Save** to save the file.

7. Select **File > Exit** to exit the text editor.

Creating a Group INI File

A group INI file contains a collection of FormWare users to whom you want to assign certain access and user rights. This allows you to control or limit the access to certain programs and functions within FormWare to only those users within a particular group. To have group rights, a user must be assigned to that group. This is accomplished by adding the user's login ID to the group INI file. Thereafter, a user logging onto Windows and FormWare with the correct login ID can access the areas of FormWare as defined by his or her group profile. To create a Group.INI file:

1. From the Administration window, open the \FormWare\INI\Group folder.

2. Right-click in the Contents area and select **New** from the context menu. A text editor displays.

3. Type the desired parameters in the file. See INI Parameters in the Supplementary Information section for a list of the parameters you can use within Group.ini files.

4. Select **Save** on the toolbar. The Save As dialog displays.

5. In the **File name** field, type the name of the group. This name must match the group's name on the system.

6. Select **Save** to save the file.

7. Select **File > Exit** to exit the text editor.

Creating a Job INI File

Job INI files contain parameters used during a Completion session for a specific job. Defining a job INI file is optional; you can specify most job properties when you define the job within the Designer module. Each job INI file bears the name of the job, with a .ini extension.

To create a Job INI file:

1. From the Administration window, open the \FormWare\INI\Job folder.

2. Right-click in the Contents area and select **New** from the context menu. A text editor displays.

3. Type the desired parameters into the new INI file. See INI Parameters in the Supplementary Information section for a complete list of the parameters you can use within job INI files. For example, you can set the following parameters:

   o The Job Files folder displays all BDFs for a job and enables you to manage the related files contained in many different directories. You must specify the FILES=<path> parameter in the job INI file to display job-related files for a specific job. The path must be the full path to the files directory, including the drive letter. When you select the Job Files folder on the File Browser, the Job Selection dialog displays. Select a job from the list and click **OK** to display all batches for that job. To display the batches for all jobs, do not select a job and select the **OK** button. All functions available in the Files folder are available in the Job Files folder.

o   The Job Images folder displays all image files for a job. You must specify the IMAGES=<path> parameter in the job INI file to display the images for a specific job. The path must be the full path to the images directory, including the drive letter. When you select the Job Images folder on the File Browser, the Job Selection dialog displays. Select a job from the list and select the **OK** button to display all of the image files in the directory specified for that job. If you want to display the images for all of the jobs, do not select a job select the **OK** button. All of the functions available in the **Images** folder are available in the **Job Images** folder.

o   Use the DFL=Y INI setting to control how forms are loaded for use within Designer and Completion. If you set DFL=Y in the System.ini or a job INI file, the forms for a job are only loaded into memory when those forms are needed within the application. This can decrease the amount of time necessary to load an application into memory when using Completion or Designer. Consider using this setting when you have are designing or running job that contains a large number of forms.

4. Select **Save** on the toolbar. The Save As dialog displays.

5. In the **File name** field, type the name of the job. This name must match the name given to the job in Designer. For example, if the job name is DEMOJOB the corresponding job INI file should be DEMOJOB.ini.

6. Select **Save** to save the file.

7. Select **File > Exit** to exit the text editor.

Note: You can also create job INI files within the Designer module. Because job INI files contain parameters used during a Completion session for a specific job, see the Editing the Job INI File section within Modifying Job Properties for Completion.

Editing the System.ini File

The System.ini file contains the set of parameters used during the Completion session. To edit this file:

1. In the Administration window, open the \FormWare\INI\System folder to display the system-related INI files.

2. Double-click on the System.ini file to open it in a text editor.

3. Type the desired parameters in the file. See INI Parameters in the Supplementary Information section for a list of the parameters you can use within the System.ini file.

4. Select **Save** within the text editor to save your changes.

Caution: Do not change the name of the System.ini file or any other INI file name within the \FormWare\INI\System folder. Doing so renders that INI file useless and can cause serious program errors or problems with program functionality.

5. Select **File > Exit** to exit the text editor.

Creating an Applets.ini File

Applets are custom programs that enable you to add and manage custom properties at the job, form, or field level within your FormWare application. This type of functionality is very useful for custom third-party applications you would like to run with FormWare, such as a custom

recognition engine or a special editing program that requires custom properties at the job, form, or field level.

Caution: FormWare makes no assumptions about what is done by the applet program, other than to reload the job (OBJECTLEVEL=JOBAPPLET) or form (OBJECTLEVEL=FORMAPPLET or OBJECTLEVEL=FIELDAPPLET) when the program returns control. The applet can use the job API and store properties in the job or form. In this case, Designer will reload and save these properties but know nothing about them. The applet can also store properties in some other file that Designer, or FormWare, knows nothing about. Therefore, it is critical to understand and carefully define the functionality of any applet before you attach it to your FormWare application.

To access any applet, you must first create an Applets.ini file:

1. Open the Administration window and display the \FormWare\INI\System folder.

2. Right-click in the Contents area to display the context menu and select **New**. An empty Editor dialog displays.

3. Type the appropriate parameters to call your custom applet. The format of the Applets.ini file is:

   ```
   OBJECTLEVEL=NAME;PROGRAM

   OBJECTLEVEL=NAME;PROGRAM

   OBJECTLEVEL=NAME;PROGRAM

   ...
   ```

4. In the **OBJECTLEVEL** portion of this parameter, specify the appropriate level to which this applet applies. Valid parameter values are:
   - **FIELDAPPLET** - Applied to the field-level applets available on the Field Properties, General tab , Component field and Configure button.
   - **FORMAPPLET** - Applied to the form-level applets available on the Form Properties, General tab, Component field and Configure button.
   - **JOBAPPLET** - Applied to the job-level applets available on the Job Properties, General tab, Component field and Configure button.

5. In the **NAME** portion of this parameter, type the text to display in the **Component** field of the Job, Form, or Field Properties, General tab. For example: **My Custom Job Settings**.

6. In the **PROGRAM** portion of this parameter, type the name of the program (applet) to launch. This program must be either fully qualified or it must exist in the search path. For example: **MYCUSTOMJOB.EXE**.

   For example, an Applets.ini file that exposes a job-level applet named MYCUSTOMJOB.EXE, and a field- and form-level applet named MYCUSTOMOCR.exe would look like:
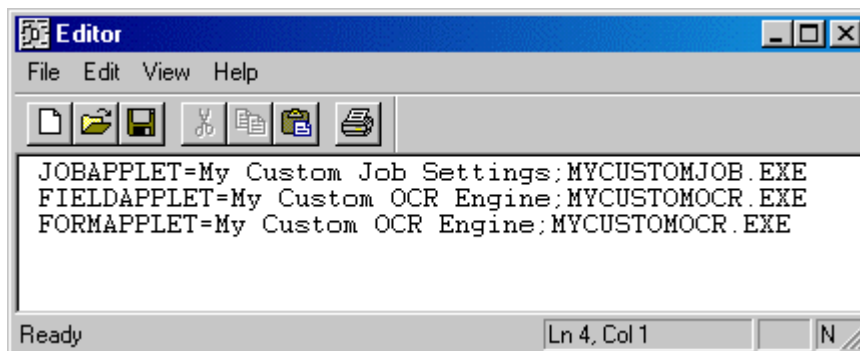
7. Select **Save** on the toolbar to save your changes to this INI file. The Save As dialog displays.

8. In the **File name** field, type "Applets" and then select **Save**. This saves the Applets.ini file to the FormWare\INI\System folder.

9. On the Administration window, select the **Designer** button on the toolbar to launch Designer.

10. Open a job in Designer and display the appropriate Job, Form or Field Properties, General tab for the object level you want. For example, here is the Job Properties, General tab:
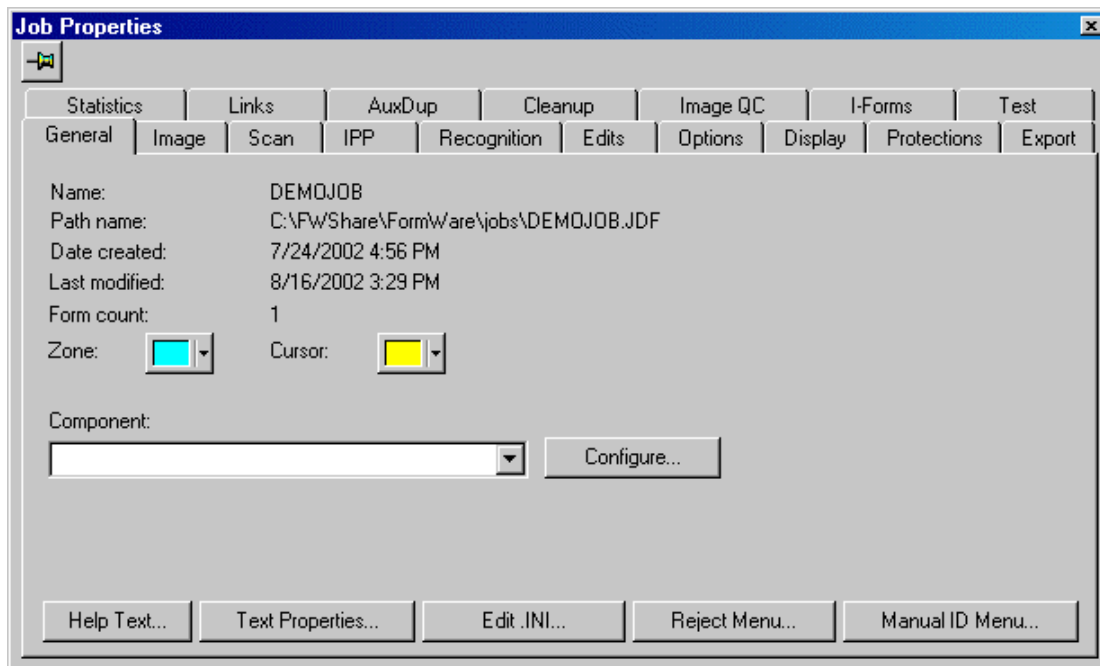
11. In the **Component** field, select the applet name from the field drop down list.

12. Select the adjacent **Configure...** button. This launches the program (EXE) you specified in the Applets.ini file. Designer launches the specified program and passes the following command line ([ ] indicates optional, depending on OBJECTLEVEL). This command line provides the applet with a context for what it is working against (i.e., job name, form name, and field name).

```
ProgramName /PROPS /JOB=JOBNAME [/FORM=FORMNAME] [/FIELD=FIELDNAME]
```

Note: If the properties need to be job, form, or field specific then the applet needs to be written to use the object level APIs and store the properties in the appropriate files, or the applet must implement its own job, form, or field settings.

13. Configure the applet as appropriate, exit, and return to Designer.

14. If the applet stores data in the job or form, select **Save All** on the Designer window toolbar to save your changes.

Note: You must configure each object level (job, form, field) that you have designated within the Applets.ini file.

Editing the Stats.ini File

The Stats.ini file, located in the \FormWare\INI\System folder, contains the programs necessary to generate operator and recognition statistic reports. You can edit the Stats.ini file to delete or

add a report. The Process menu displays the results of any changes made to Stats.ini file when you have the Statistics folder open.

Creating a Titles.ini File

A Titles.ini file can change the title information displayed at the top of a window when you run a FormWare module. Set the name of the FormWare program (without the EXE extension) equal to any string in this file. This string can be alphabetic or numeric and up to 32 characters. The strings must be unique for all modules listed in the file.

For example, you can change the window title bar information for the Completion module, by typing the following in the Titles.ini file: COMPLETE=HCFA Processing Software. Now, when you run the Completion module, the window title bar displays "HCFA Processing Software."

You create and store this INI file in the \FormWare\INI\System folder.

Creating a TabPrep.ini File

You can create a TabPrep.ini file to include a custom subset of the recognition engines available in the **Table to Generate** list in the Table Preparation dialog in Designer: CGK RSP AEG CMP INS NST PAR.

You can also edit an existing TabPrep.ini file to eliminate a recognition engine from the **Table to Generate** list. If the TabPrep.ini file does not exist, all recognition engines will display on the Table Preparation dialog.

The TabPrep.ini is stored in the \FormWare\INI\System folder.

Creating Jobs

A job contains one or more data forms, or Form Definition files (FDF). A data form is a collection of fields that defines the data you want to extract from an image or paper form. You create the data form, including all fields and labels on the data form, using the Designer module. The data form displays on the screen during the completion process.

The Job Definition file (JDF) contains the high-level information about the job. A JDF is a collection of one or more FDFs with additional job-specific attributes created with the Designer module. JDFs are located in the \FormWare\Jobs folder and have the job name with a .jdf extension. You can copy, rename, delete, or create jobs and job help files using functions accessed through this folder.

A job help file is a text file available to the Completion operator. During a Completion session, the operator presses a key to view the file. This file can contain as much information as necessary to help operators as they process the job. Job help files are also stored in the \FormWare\Jobs folder.

The first step in designing your forms processing application is to create a new job. When you define a new job, the system creates a new JDF.

Creating a New Job

To create a new job within Administration:

1. From the Administration window, select the **Designer** button. The Designer window displays.

2. Select the **New** button on the Designer window. The New Job Name dialog displays.

New Job Name

Enter the name of the new job to create.

OK

Cancel
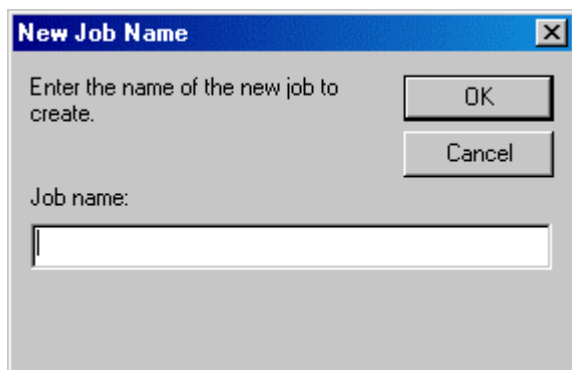
Job name:

3. In the **Job name** field, type a descriptive name for your job. FormWare will create a new JDF with this name and a .jdf extension. Select **OK**. The Job Defaults dialog displays.

Modifying Job Default Properties

Properties are associated with each object within a FormWare job. There are three basic types of objects within the Designer module: Field objects, Form objects, and Job objects. To access the properties for an object, double-click on the object in the Job Browser of the Designer window. Each property dialog contains tabs where you set specific properties for the active object.

To anchor the Properties dialog, click on the **Push pin** button in the upper left corner. With this button toggled on, the Properties dialog remains visible as you select different objects within the Designer module and displays information specific to the selected object.

You will use the Job Defaults dialog to modify for the job any of the properties you set on the System Defaults dialog. The defaults apply globally to the job, and to the forms and fields within the job. Some of these settings can later be overridden at the job, form, and field property levels within the main Designer window.

1. Create a new job. Once you name the job, the Job Defaults dialog displays.

2. Change any property settings that you want to be different than you set on the System Defaults dialog. For descriptions of these fields, refer to the Modifying System Default Properties section.

Note: The Job Defaults dialog does not include **Zone Type** options or the **Font** selection field that do exist on the System Defaults dialog.

3. Select **OK** to create the new job. The Designer module window displays the new job. At this point, the job contains an empty Image Viewport and a blank data form.

4. To see the job properties, double-click the job name in the Job Browser. The Job Properties, General tab displays the following information about the job:
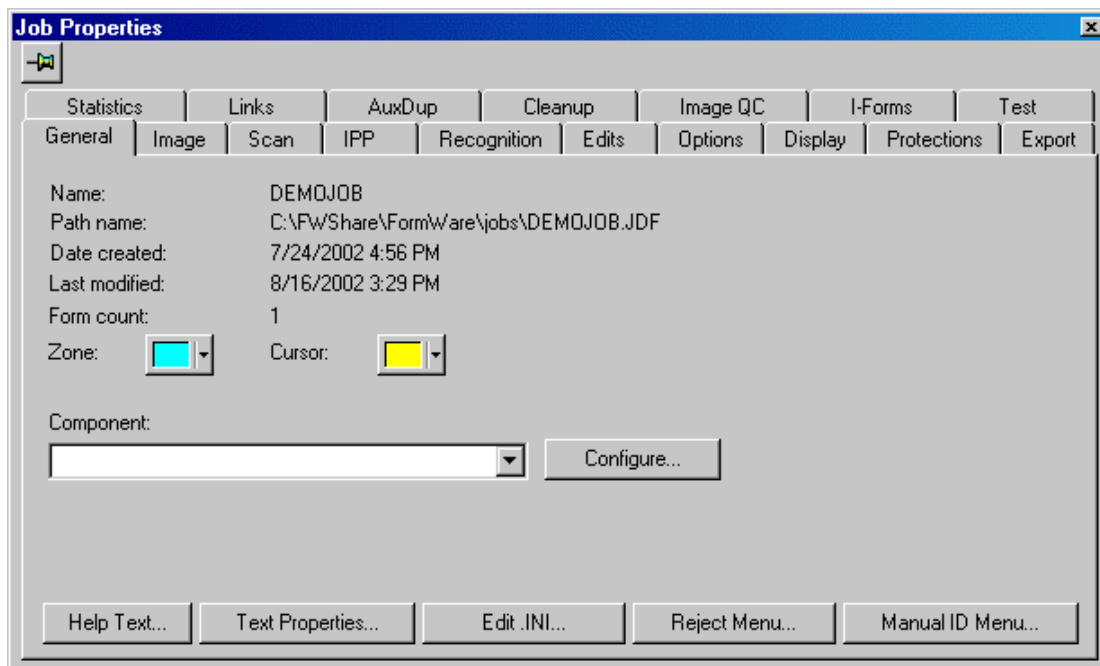
- o **Name** – Name of the current job or JDF.
- o **Path name** – Directory location of the current JDF.
- o **Date created** – Date and time the JDF was created.
- o **Last modified** – Date and time the JDF was last saved.
- o **Form count** – Number of data forms (FDFs) associated with the current job.
- o **Zone** – Background color applied to recognition zones during a Completion session. Select the arrow next to this field to select a different color or create your own.
- o **Cursor** – Background color applied to the active reject character (i.e., current cursor location) on both the data field and the recognition zone during a Completion session. Select the arrow next to this field to select a different color or create your own.

5. In the **Component** field, select from the list of export, recognition, or other third-party components that need configuration at the job level. The options available in this list depend on the settings in the Applets. ini file. Refer to the Creating an Applets.ini File section for more information. Select the **Configure** button to display the configuration dialog for the component you selected.

6. Select the Image tab to define image properties for the job. The Job Properties, Image tab displays.
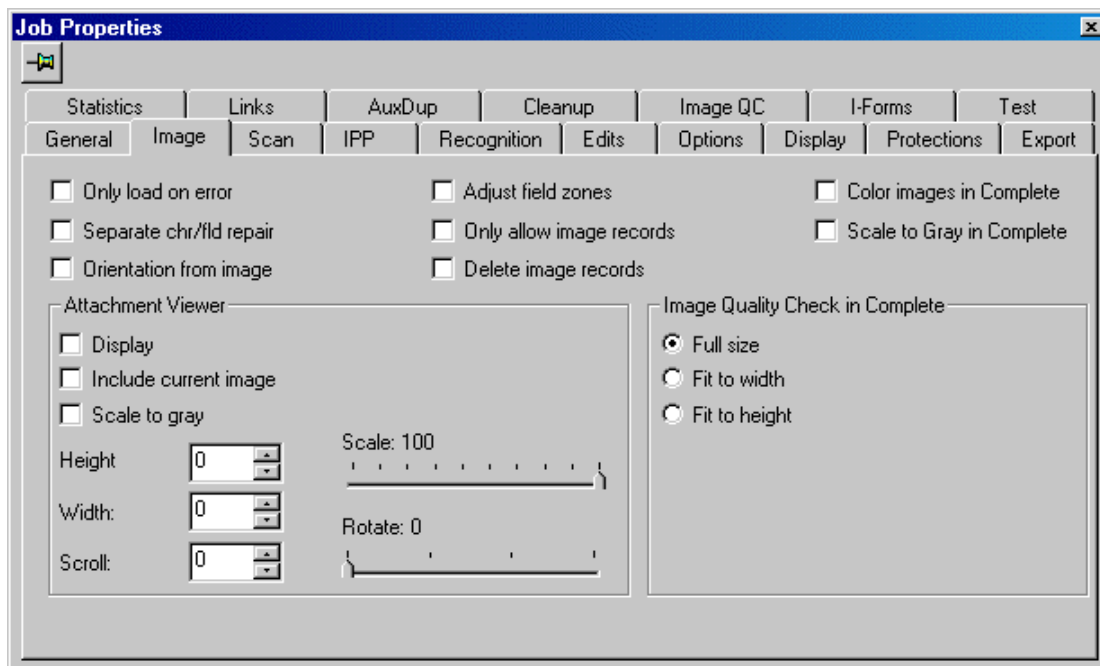
7. You can select or deselect from the following image options:

    o **Only load on Error** - Loads only those images that contain a character or field reject display. This feature is designed for jobs that are 100% recognition applications. Any Key from Image (KFI) fields (i.e., no recognition is performed) are ignored or skipped by the Completion process if you have selected this option.

    o **Separate chr/fld repair** - Converts field reject characters to error flags. Operators can repair error flags in the Remove Flags mode of Completion. Operators performing reject repair are presented only the character rejects. All field rejects become flagged fields and are not presented to the reject repair operator.

    o **Orientation from image** - Checks for orientation flags (i.e., image rotation) in TIFF image headers. By default, the Completion module ignores the orientation flag in the TIFF header.

    o **Adjust field zones** - Indicates to the Completion module to adjust the zone coordinates according to the information returned from the recognition engine, rather than using the zone coordinates created within Designer.

    o **Only allow image records** - Allows only image records in the batch. This option prevents Completion operators from adding records without images to the batch.

    o **Delete image records** - Allows the Completion operator to delete a record that has an associated image. By default, an operator cannot delete a record that is associated with an image.

    o **Color images in Complete** - Indicates to Completion to use color images, if available.

    o **Scale to Gray in Complete** - Indicates to Completion to use multiple shades of gray to *smooth* the image when displayed.

Note: You will set the Attachment Viewer and Image Quality Check options on this tab during other tasks.

8. Select the **Save All** button on the Designer window to save your property settings.

Working with Data Forms

Data forms contain the fields where data is inserted during data completion. By creating a data form, you create a Form Definition file (FDF). An FDF is a collection of fields placed on a data form and contains the display, recognition, and completion information for the data form. You use the Designer module to create FDFs. FDFs are saved in the \FormWare\Forms folder along with the form help files used during Completion.

When you create a job, one blank data form is automatically created with the job and has the same name as the job. This data form is listed in the Job Browser area of the Designer module window beneath the Forms folder.

Caution: Be very careful when deleting or renaming data forms. An FDF is part of a Job Definition File (JDF). If the FDF is deleted or renamed, the JDF functions incorrectly and you receive an error.

A form help file is a text file that is available to the Completion operator. During the Completion session, the operator presses the [F3]-[Esc] keys to view the form help file. This file can contain as much information as desired. It is available only when the operator is using the associated FDF.

The \FormWare\Forms folder also contains the setup image used to design the FDF and the recognition results file from a recognition test on an image.

The FormWare system default enables you to define up to 98 forms per job. However, you can increase this default by adding the FEX=Y parameter in either the System.ini file (for all jobs within the FormWare system) or the Job ini file (for a specific job). Adding the FEX=Y parameter to the appropriate INI file increases the number of forms allowed to a maximum of 16,000 per job.

Creating a Data Form

Follow these steps to create a new data form (FDF):

1. Open a job in the Designer window.

2. Select the **New** button. The New window displays.

3. Select Form and then **OK**. The New Form Name dialog displays.

4. In the **Form name** field, accept the default or type a new name and select **OK**. The new data form displays in the Designer module window, and the icon displays beneath the Forms folder in the Job Browser.

Note: Before you can modify the data form, the Form Properties dialog for that data form should display. If it does not, double-click on the form name in the Job Browser.

5. Add one field to the form by selecting a field type from the toolbar and clicking a spot on the data port for the form. Some testing processes require at least one field to be on the form. Later you will learn how to add and define specific types of fields.

6. Select **Save All** on the Designer window toolbar to save your changes.

Setting Form Properties

Use the Form Properties dialog to specify the properties for an entire data form.

1. Open the job with the form in Designer.

2. Double-click on a form name in the Job Browser to display the Form Properties dialog. The General tab displays the default information about the form (e.g., name, path, create and modify dates).
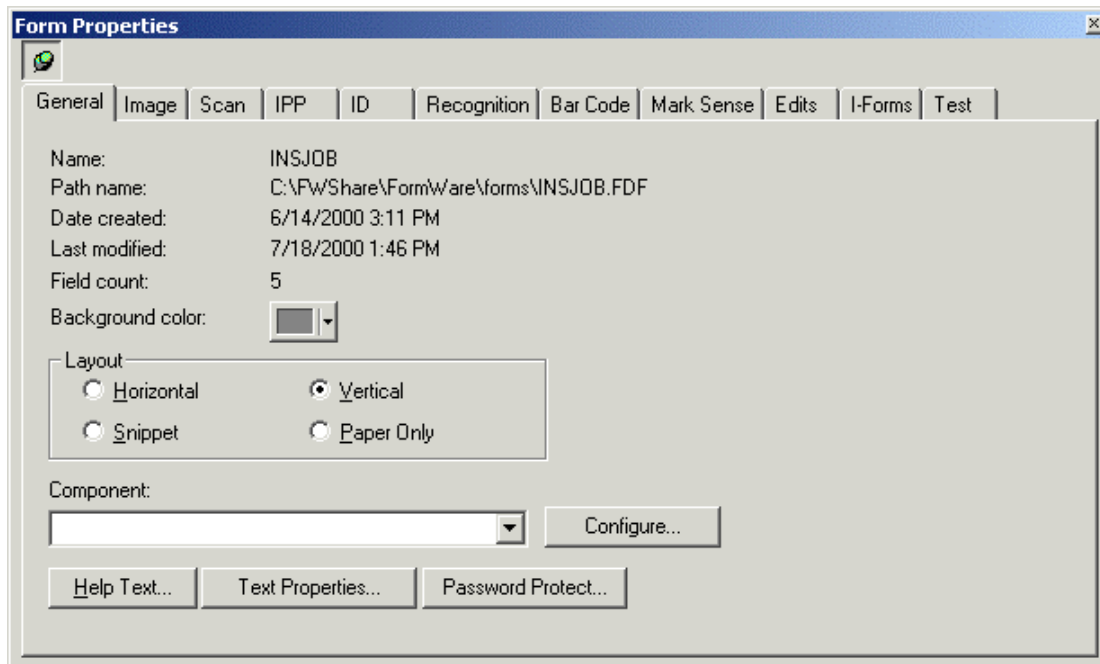
3. Select the **Pushpin** button in the upper-left corner of the dialog to keep the dialog active.

4. On the General tab, select the arrow in the **Background color** field to select a new color for the form background.

5. In the **Layout** area define the form orientation for this form by selecting the appropriate radio button. Some projects require both horizontal and vertical orientation. Designer enables you to convert vertical to horizontal orientation and vice versa on a per form level. You might need to adjust and modify the zones, using hot keys in the attachment viewer.

6. In the **Component** field, select from the list of export, recognition, or other third party components that need configuration at the form level. The options available in this drop-down list depend on the parameters in the Applets.ini file. Refer to the Creating an Applets.ini File section for more information.

7. Select the **Configure** button, to define the component you selected.

8. Select the **Text Properties** button to display a text editor that contains form-level, extended FormWare properties, and properties for third-party software used with FormWare. Although you can view and modify these values from the editor, we recommend this for FormWare advanced users only. Currently, the form-level export component and I-Form settings are stored in this text file.

9. To create a help file using the **Help Text** button, refer to the Creating a Form Help File section.

10. To password-protect the form using the **Password Protect** button, refer to the Protecting a Form with a Password section.

11. Select the **Image** tab on the Form Properties dialog to set properties specific to the images associated with the data form.
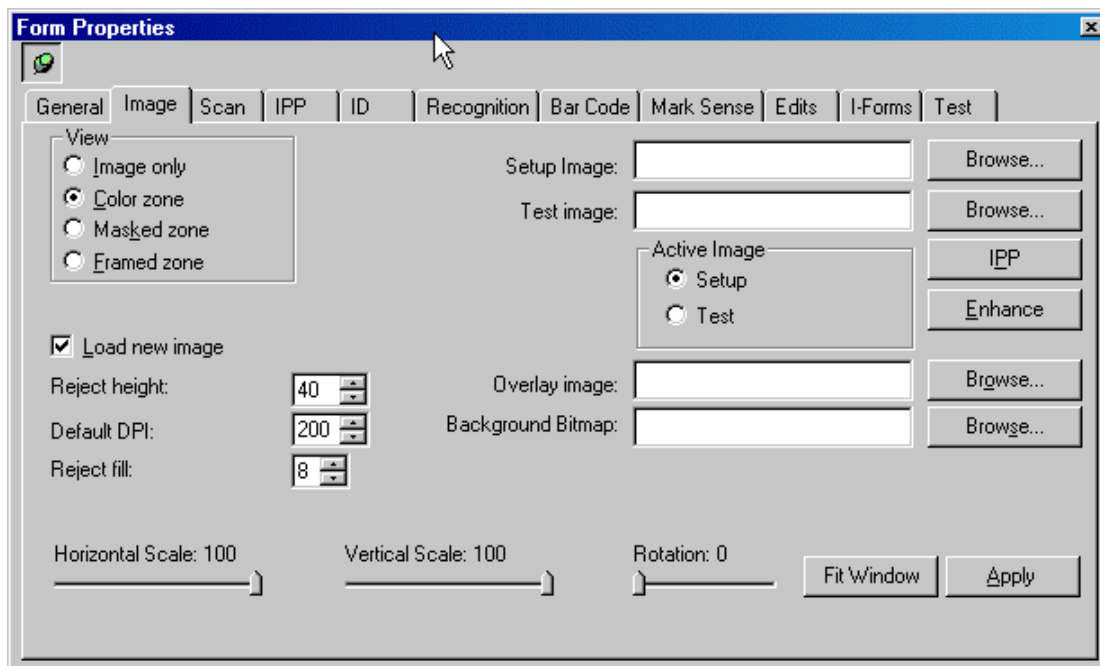
12. In the **View** area, select the option that specifies the display of the data form and image. Options are:

    o **Image Only** - Displays the image in the viewport (cannot be used with Recognition).
    o **Color Zone** - Displays the image in the viewport and highlights the recognition zone with the color designated on the General tab of the Job Properties dialog. You can use this option for both Recognition and KFI applications.
    o **Masked Zone** - Masks the image and displays only the recognition zone. You can use this option for both Recognition and KFI applications.
    o **Framed Zone** - Displays the image and places a thin red line around the recognition zone. You can use this option for both Recognition and KFI applications.

13. See the Selecting a Setup Image section for instructions on using the **Setup Image** field and the Specifying an Overlay Image section for instructions on using the **Overlay Image** field.

14. See the Setting Form Removal Properties section for instructions on using the **Test Image** field.

15. Select the **Browse** button next to the **Background Bitmap** field to display the contents of the \FormWare\Bitmaps folder where you can browse for a JPEG file to use as the background bitmap image for this data form during Completion. This file must be a JPEG (.jpg) file.

16. Select the **Load new image** check box to load a new image with the start of each new record in a Completion session. When the data is completed for a record, the next image in the BDF is displayed with a new record. When you deselect this option, the operator can manually step through the images in the BDF. The operator can also input any number of records per image.

    In Completion, a record is created of the data extracted from a data form. The data form can have a one-to-one ratio with an image, or one image can create multiple records. However, one record cannot be created from multiple images. If you do not select the Load new image check box, the operator must load a new image from the Completion module Image menu. Data forms are presented in the Completion module based on form linkage or form identification information in the BDF, not based on images. You set form linkage on the Links tab on the Job Properties dialog.

The **Load New Image** option also affects Recognition, especially the InScript Recognition engine and recognition zones on the data form. The InScript recognition engine currently has a limit of 256 zones per form. However, the InScript engine counts multi-mark sense boxes, paragraph zones, and table zones as only a single zone, although it actually encompasses numerous fields. With this information, you can design your forms to group several of these zones together, thereby eliminating this zone limitation. However, some applications might not be designed in such a way that this grouping makes sense. Under those conditions, you can distribute the zones across multiple forms. To achieve this, deselect the **Load New Image** check box on all forms except the first form. This causes the recognition engine, as well as other processes, to use the same TIFF image with multiple form definition files, until it reaches a form that is designated with the **Load New Image** option.

17. In the **Reject height** field, select the height in pixels of the reject characters displayed in the ticker tape during a Completion session. You can see the height of the reject characters increasing in the ticker tape display of the Designer module.

18. In the **Default DPI** field, select the dots-per-inch (DPI) of the image being processed if the information is not contained in the image header. When you use recognition technologies, many engines must know the DPI of the images being processed. If you have images that do not contain the DPI value in the TIFF header, specify the DPI in this field.

19. In the **Reject fill** field, select the space in pixels between reject characters displayed in the ticker tape during a Completion session.

20. Set the magnification of the image in the Image Viewport using either:

    a. The **Horizontal Scale, Vertical Scale**, and **Rotation** fields to manually adjust the size of the image in the Viewport.

    Or,

    b. The **Fit Window** button, which automatically resizes the image in the Viewport to fit completely within the Viewport window.

21. Select **Apply** to accept your changes and return to the Designer window to view your changes to the image in the Image Viewport.

22. Select the Edits tab on the Form Properties dialog.

23. Select the **Start of Document** check box to identify the first page, or data form, in the document. This option is not required for form identification purposes, but is required for other document-related options to work correctly. During a Completion session, when a main document form is encountered, the document count is incremented. This count is independent of the total number of data forms processed, and instead relies on the data form specified as the Start of Document. The **Document Name** field becomes active when you select this option.

Note: A document is a multi-page form that you want to group together for forms processing purposes. When you set up a document application, you must designate which data form is the main form, or the first page of the document. To do this, select the **Auto ID Linked Forms** checkbox on the ID tab on the Form Properties dialog.

---

24. In the **Document Name** field, type the name of the document, which then applies to all forms related to the document (e.g., first page and attachments). This applies to all forms that are linked, not necessarily all forms in the job.

25. In the **Edit DLL** field, type the DLL name or browse to locate an existing DLL. Programmers can create a DLL for the data form that executes during a Completion session. Put all DLLs in the \FormWare\Programs folder.

26. In the Rule File field, type the Rule file name or browse for an existing Rule file. Rules are applied to data for verification purposes. Rule files must be located in the \FormWare\Rules folder. To edit the rule file, select the **Rules** button where you can modify the rule in the Rule Editor.

27. Select the **Attachment** check box to indicate that this data form is an attachment. Attachments are images used during a Completion session that are informative to the operator. Attachments can also be viewed from other images from which data is being extracted. See the Setting Up the Attachment Viewer section for more information about defining attachments.

28. Select the **Save All** button to save your form property settings.

Protecting a Form with a Password

You can protect individual forms within a job. Once protected, you must supply the password before you can modify the form.

Caution: If you lose the password, you can no longer modify the form. Therefore, be sure you record all form passwords you create and store them in a safe place.

To create a password for a form:

1. Open the job with the form you want to protect in Designer.

2. From the Job Browser, double-click on the form name to display the Form Properties dialog.

3. On the General tab, select the **Password Protect** button. The Form Password window displays.

4. In the **Password** field, type a password and then record it in a safe place.

5. In the **Verify Password** field, re-type the password.

6. Select **OK** to save the password and return to Designer.

7. To exit the password dialog without creating a password, select the **Cancel** button. You can also select the **Clear** button and confirm that you want to erase the password. Then, select **OK** to return to Designer.

Creating a Form Help File

To assist your operators, you can create help files for each form in the job or just one for the job. To create a help file for a specific form:

1. In Designer, open the job that contains the form for which you want to create help.

2. Double-click on the form name in the Job Browser to display the Properties dialog for the form.

3. On the General tab, select the **Help Text** button. The Editor dialog displays.

4. Type your help text.

5. Save and close the editor. The file is saved in the \FormWare\Forms folder with the name of the form and an .hlp extension.

6. Select the **Save All** button to save the help text within the job.

Editing a Form

To edit a data form (FDF) from the Administration module:

1. Select the Forms folder in the File Browser.

2. Double-click on the FDF you want to edit in the Contents area of the Forms folder to launch Designer. Within Designer, only the data form displays, not related jobs.

3. Modify the data form as necessary.

4. Save the form and close Designer to return to Administration.

Changing a Form Name from the Job Browser

To quickly change the name of an existing form within a job:

1. Highlight the name of an existing form within the Designer window Job Browser.

2. Right-click to display the context menu and select the **Rename** option.

3. Type the name of the new form. This name must be unique among the other form names within the job. You may want to refer to this form in a rule so make sure that the name you use is identifiable. Valid characters for a form name are A-Z, a-z, 0-9, _, and -. Any other character is unacceptable and will display an error message.

4. Select **Enter.**

5. Select the **Save All** button on the toolbar to save your changes.

Importing a Data Form

Using the data form import function, you can import a data form from another job into the current job. This feature saves you the time of recreating fields and labels that are identical between two forms.

If you are importing a form that was previously exported, you may need to adjust the scale factors. Before exporting, you must set the scale factors to 100%. After import, you can change them back.

To import a data form:

1. In Designer, open the job into which you want to import a form.

2. Right-click on the Forms folder in the Job Browser to display the context menu.

3. Select the **Import** option to display the submenu. Options depend upon the software you are licensed to use (e.g., FormWare, EZC, FormWare Export).

4. Select the import option for the appropriate file type from the submenu. The Open dialog displays the folder where forms of the type you selected are located (e.g., \FormWare\Forms or \FormWare\Jobs).

5. Select a form to import and select the Open button. You can also navigate to another folder to find the form you want to import. The form is imported into the current job.

6. Right-click on the form to select the task you want to perform. You can cut, copy, delete, or rename the form. You can also cut and copy fields from one data form to another.

Caution: An imported data form is **not** a copy of a data form—rather, it is the actual Form Definition file (FDF) you selected. If you change the name or contents of the FDF, it can impact other jobs that use the same FDF.

When you import a form from a source other than FormWare, only those properties that are compatible with FormWare are imported. For example, an InScript FDF does not contain information on field lengths or field starting positions. Therefore, when you import an InScript FDF into FormWare, the resulting data form contains fields that all have the same length and are positioned on the left edge of the form. You must assign new lengths and positions.

Exporting a Data Form

When you export a form in Designer, if the image has been scaled to something other than 100%, you must reset the scale factors to 100% before exporting. After the export, set the scale factors back to the desired values. Additionally, if you import a FormWare Export file, and the above logic was applied before exporting, then you need to set the scale factors on the imported form.

To export a data form:

1. In Designer, open the job from which you want to export a form.

2. Right-click on the form name in the Job Browser to display the context menu.

3. Select the **Export** option to display the submenu. Options depend upon the software you are licensed to use (e.g., FormWare, HTML).

4. Select the option for the appropriate file type from the submenu. Select from these options:

    o **FormWare** - Exports the form to a binary text (TXT) file A message displays indicating that the TXT file has been created and provides you with an option to view the file. Exported TXT files are placed in the \FormWare\Forms folder.

    o **HTML** - Displays the Export HTML dialog where you can type the path and filename to which you want to export the file.

Defining Multiple Forms Per Image

When an image contains more than one form, you can divide the forms among multiple FDF files. You can also create different data forms for different sections of the image. For example, you can create one data form to extract address information and another to extract financial information from a single image.

To specify multiple forms for a single image:

1. In the Designer Job Browser, double-click the first data form to specify for the image and display the Form Properties dialog.

2. Select the Image tab.

3. Select the **Load New Image** check box.

4. Open another data form within the job and display the Image tab on the Form Properties dialog for that form.

5. Deselect the **Load New Image** check box for that form. By disabling this option, you force the form to use the image assigned to the first form in the job.

6. Select the **Save All** button on the Designer window to save your changes.

Selecting a Setup Image

Any job that is not a key-from-paper application requires a setup image that is the basis for your data form design. If you have more than one form within your job, each form can have its own setup image. This image must be either a JPEG or TIFF image saved with Group-4 compression. The following sections contain instructions for identifying and adjusting a setup image.

Scanning a Setup Image

When you have a scanner attached to your workstation, you can scan the setup image directly from the Designer module.

Note: Before you can proceed with scanning, you must set the scanner properties. See the Defining Scan Settings section for more information.

To scan a setup image:

1. From Designer, double-click the job name in the Job Browser to display the Job Properties dialog.

2. Select the **Push Pin** button to maintain the display of the Job Properties dialog.

3. Display the Job Properties, Scan tab and select the **Properties** button to open the Scan Designer module for the current job.

4. Select the **Open** button to choose the appropriate scan configuration file (SCG) for the job. If you have not defined a scan configuration file yet, see the steps in the Defining Scan Settings section.

5. When you finish setting the scan properties for your job, select **OK** to return to the Job Properties dialog.

6. Double-click on a form name in the Job Browser to display the Form Properties dialog.

7. Select the Form Properties, Scan tab.

Note: The **Test** button on this tab is always enabled. However, if you do not have a scanner attached to your workstation, you will receive an error message within the Scan Operator.

8. To test your scanner properties, insert a paper document into the scanner and select the **Test** button on the Scan tab.

9. The document is scanned and the resulting image displays in the Image Viewport on the Designer window. If there is a problem with the scanned image, modify the settings in Scan Designer until you achieve the desired results.

Specifying a Setup Image

The setup image is your reference image when designing the data form and fields. This image should be representative of the paper forms processed with the current data form. To specify a setup image for a form:

1. Open a job in Designer and double-click on the form name in the Job Browser to display the Form Properties dialog.

2. Select the Image tab.
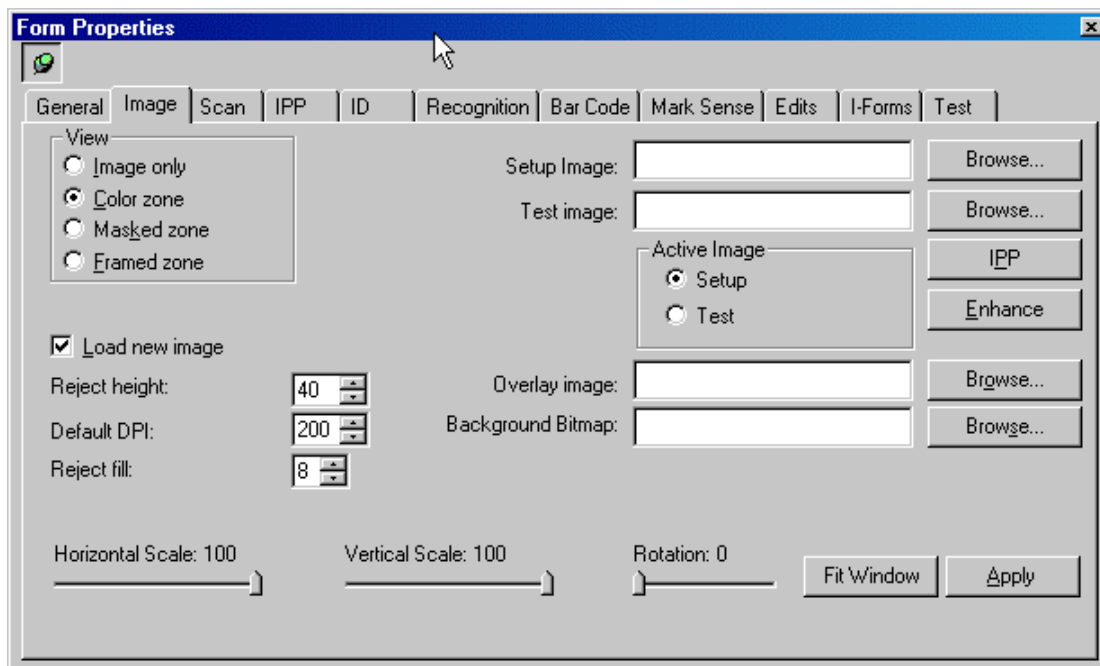
3. Select the **Browse** button next to the **Setup Image** field and locate the file for the image you want to use as the setup image. Once you select an image, it is copied to the FormWare\Forms folder and renamed to the form name with a .sup extension (e.g., Demo.sup).

4. Select the **IPP** button to run Image Pre-Processing (IPP) on the setup image. The results display in the Image Viewer. Select **OK** to save the IPP changes to your setup image or **Cancel** to reject the changes. For instructions on setting IPP properties see the Setting Image Pre-Processing Properties section.

5. Select the **Enhance** button to run the Enhancement process on the setup image. The results display in the Image Viewer. Changes occur only if you have registered the image first. See the Registering an Image section for more information. You must also have one field on the form before Enhancement can run. See the Enhancing an Image section for instructions on defining the Enhancement process.

6. Select the **Save All** button to save your setup image in the job.

Changing Setup Image Magnification

To change the magnification of the setup image within the Image Viewport on the Designer window:

1. Follow the steps to Specify a Setup Image for your form.

2. On the Form Properties, Image tab use the **Horizontal Scale**, **Vertical Scale**, and **Rotation** fields to manually adjust and rotate the setup image within the Image Viewport on the Designer window.

   Or,

   Select the **Fit Window** button to automatically resize the image to fit completely in the current Image Viewport.

3. Select **Apply** to accept your settings and return to the Designer window to view your changes.

4.   Select **Save All** on the Designer window to save your changes.

Adjusting the Setup Image

After you retrieve a setup image, you need to prepare it prior to designing your data form. The setup image needs to accurately reflect the production images after they have been processed.

You can change the section of the setup image that displays in the Image Viewport:

1.   On the Designer window, click and hold your right mouse button in the Image Viewport.

2.   When the cursor changes to a hand, drag the mouse across the Image Viewport to move the image.

3.   When the appropriate area of the image displays in the Image Viewport, release your mouse.

4.   Select the **Save All** button on the Designer window to save your changes.

Specifying an Overlay Image

An overlay image acts as a transparent template of the form. The overlay image is placed over the form image during Completion to help the operator determine what data should go in the fields. You create overlay images using a blank version of the paper form that is scanned with a bulb that does not cause the form lines and boxes to drop out.

To specify an overlay image:

1.   Open a job in Designer and double-click on a form name within the Job Browser to display the Form Properties dialog.

2.   Select the Image tab.

3.   Select the **Browse** button next to the **Overlay Image** field to open the \FormWare\Images folder and select an image file (TIF) to use as the overlay image for this form. Once you select an image, it is renamed to the form name with a .ovl extension (e.g., Demo.ovl) and stored in the \FormWare\Forms folder.

Note: The image you specify as an overlay image must be pre-processed prior to selection. If you specify horizontal and vertical scaling factors, the overlay image will also be scaled to this factor.

---

4.   Set the magnification of the overlay image in the Image Viewport using either:

   a.   The **Horizontal Scale, Vertical Scale**, and **Rotation** fields to manually adjust the size of the image in the Viewport.

   Or,

   b.   The **Fit Window** button, which automatically resizes the image in the Viewport to fit completely within the Viewport window.

5.   Select **Apply** to accept your changes, return to the Designer window and view your results.

6.   Select **Save All** on the Designer window to save your changes to the job and form.

Adding Fields to a Form

A data field is an individual container that captures specific information from an image. This occurs when you use recognition technology or when an operator types directly into this container on the data form. Each field on the data form has its own special characteristics called properties. You can set field properties to determine the size of the field, type of information that can be contained in the field, and other criteria.

Selecting a Field Type

Follow these steps to create a new field on your data form. You can create a data field by selecting a button from the Field Types toolbar, by right-clicking on the form to display and select a field from the field types list, or by clicking on the setup image, and dragging a recognition zone.

To add a field to a form:

1. Open a job in Designer and select a form within the job. The Designer window displays. The data form area (default location is the lower left portion of the Designer window) is where you will insert fields and labels.
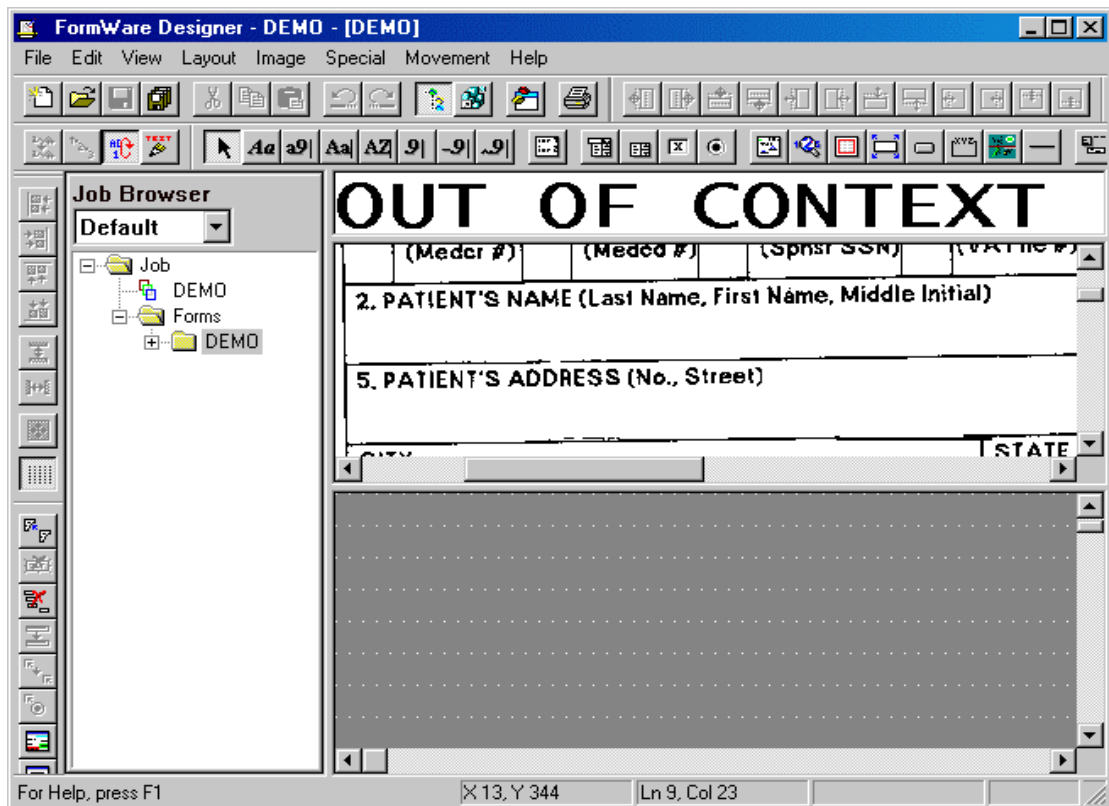
2. There are two ways to insert a new field on the data form:

    a. Select the appropriate button on the **Field Types** toolbar and then click on the data form to anchor the field and drag the cursor to size the field to the desired length. Make certain that your cursor does not go above the top grid line so that the field does not snap to the wrong grid line.

b. With your cursor on the data form, right-click to display the context menu and select the
    New Field option. The Field Types context menu displays the list of available field types
    you can use.
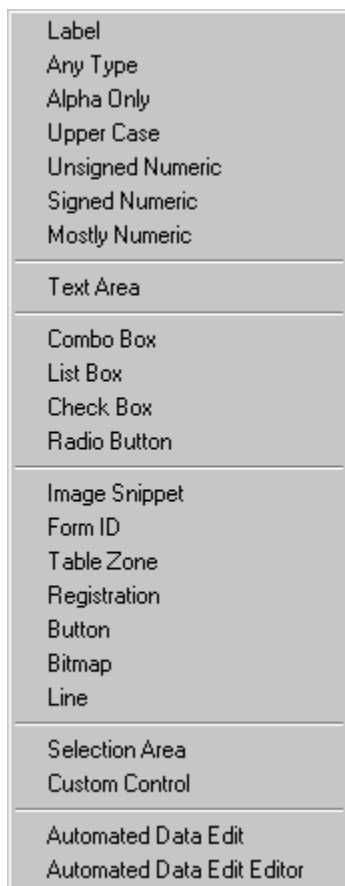
Field types include:

o   **Label** - Creates a field label to use with other field types (e.g., combo boxes, check
    boxes, radio buttons, entry fields). Label fields support multiple lines of text up to
    1024 characters in length. You can size both the width and height of a Label field.

o   **Any Type** - Creates a data field that accepts alphanumeric characters. This includes
    the entire ASCII character set except the backslash (\), which acts as a field release.

o   **Alpha Only** - Creates a data field that accepts only alphabetic characters.

o   **Uppercase Only** - Creates a data field that accepts only uppercase alphabetic
    characters.

o   **Unsigned Numeric Only** - Creates a data field that accepts only the numeric
    characters 0-9, but does not accept any sign characters, such as a minus sign.

o   **Signed Numeric Only** - Creates a data field that accepts only the numeric characters
    0-9 and the minus sign.

o   **Mostly Numeric** - Creates a data field that accepts alphanumeric characters,
    including sign characters.

o   **Text Area** - Creates a multi-line input field that accepts text up to 2048 characters in
    length. You can size both the width and height of a Text Area field.

- o **Combo Box** - Creates a combo box (drop-down selection list) field. The items that display in the combo box are determined by the contents of the LST file you select for the field. Combo Box fields can facilitate I-Form hot links.
- o **List Box** - Creates a list box field. The items that display in the list box are determined by the contents of the LST file you select for the field. List Box fields can facilitate I-Form hot links.
- o **Check Box** - Creates a check box field.
- o **Radio Button** - Creates a radio button field. More than one radio button must exist to enable a user to select from multiple options.
- o **Button** - Creates a button. How a button displays in your browser can vary depending on your computer's internal capabilities. Within Completion, a button should look essentially the same as it does when you create it in Designer.
- o **Frame** - Creates a frame. For example, you can use a frame to create a box around a group of fields or around a set of radio buttons. Frames are not applicable for I-Forms and are viewable only in Completion.
- o **Bitmap** - Creates an area for bitmap file insertion.
- o **Line** - Creates a horizontal line at the point of insertion on the form.

Note: You can also add a field to a form by creating a recognition zone on the Image Viewport first.

Setting Field Properties

Use field properties to specify options associated with a field. Once you create a field, you can change the field type, specify a field name, or further define the field properties. To do this:

1. On the Designer window, add a new field to the form.

2. Double-click on the field to display the Field Properties, General tab.

3. Select the **Push Pin** on the Field Properties dialog (upper left-hand corner of the dialog) to keep the properties dialog in focus.

4. In the **Name** field on the General tab, type a unique name for the field. You may want to refer to the field in a rule so make sure the name is identifiable. Valid characters for a field name are A-Z, a-z, 0-9, _, and -. Any other character is not acceptable in a field name. The number assign to the field based on creation order displays in the **Number** field.

5. If you are defining a Label, type the text for the label in the **Text** field.

6. The **Type** field displays the field data type you selected. To change the type, select a new type from the list.

7. The **Ln** (Line) and **Col** (Column) fields display the row and column location of the field on the data form. The **Size** field, displays the current size of the field in number of characters. You can resize the field with your mouse or use the arrows next to this field to change the field size.

8. In the **Display Mode** field, you can select a mode number to assign this field to a specific pass during a multi-pass application. See the Creating a Multi-Pass Application section for more information.

9. In the **Help Text** field, type the help text that will display on the status line when the cursor is on this field during Completion.

10. In the **Background Color** field, select a color to display in the background of this field.

11. In the **Text Color** field, select a color for the field data text.

12. If you are defining a Label, select the **Font** button to select from the list of available fonts, styles, and point sizes. You can also set other options, including color. These settings determine the font for the current label only. The name of the font displays in the **Font Name** field. To return to the font selected on the System Defaults dialog for the job, select the **Reset** button.

13. In the **Component** field, select from the list of export, recognition, or other third-party components that need configuration at the field level. The options available in this drop-down list depend on the parameters in the Applets. ini file. Select the **Configure** button to define the component. Refer to the Creating an Applets.ini File section for more information.

14. Select the **Text Properties** button to access extended FormWare properties and properties for third-party software to be used with FormWare. These properties are stored in text-based field properties, and although you can view and modify these values from the editor, we recommend this for advanced FormWare users only.

15. Display the Field Properties, Checkbox Edits tab for the field. The settings on this tab control the verification of the data field, the automatic functions, and other edits executed on the data field.
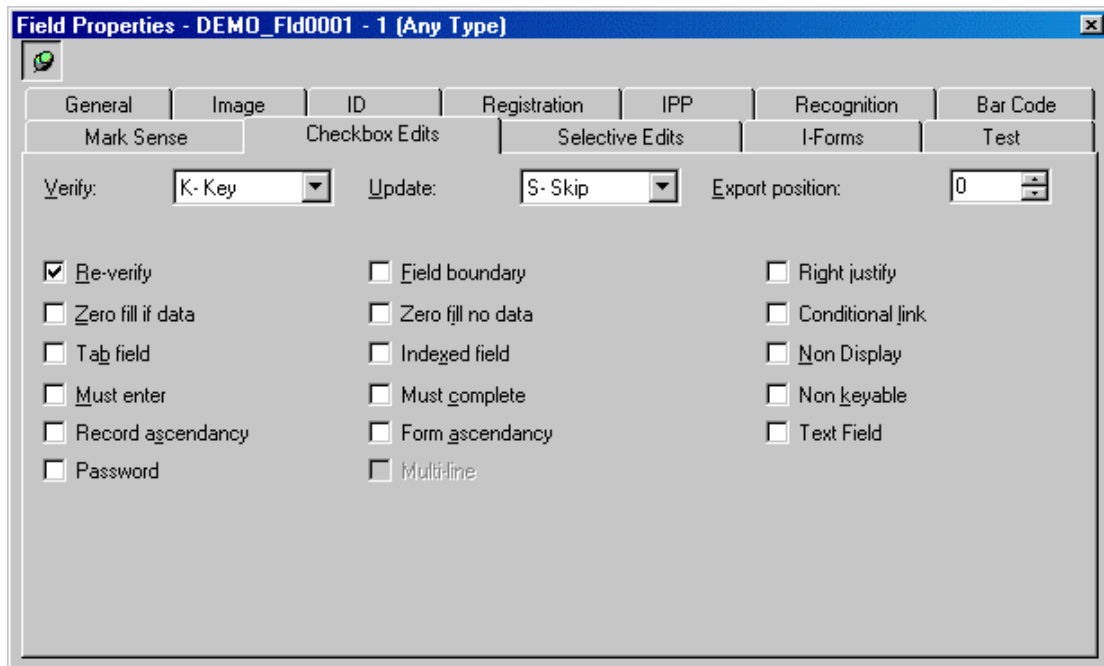
16. In the **Verify** field, select the option that sets the behavior for the data field when the Completion module is run in Verify mode. Options are:

- o **S (Skip)** - Skips the data field (i.e., the cursor does not move to the data field in Verify mode).
- o **K (Key)** - Requires that an operator key the data. Typically, this is *blind* verification, meaning that the verify operator cannot see what the entry operator has keyed. As an entry operator types each character into the data field, it is compared to the existing character. If there is a discrepancy, the verify operator receives an error message.
- o **E (Enter)** - The cursor stops on this field for every record and allows the entry of data in a previously blank data field.
- o **C (Acc not 0) 0**- If the accumulator for this data field is not equal to zero, the cursor stops on it to be key-verified.
- o **V (Visual)** - Requires visual verification. If you want to use the parameter QCO=Y (Quality Check rejects only), you must set the field type to Visual verify.
- o **R (Release)** - Ignores all other verify options and releases the record.

17. In the **Update** field, select the option that sets the behavior for the data field when the Completion module is run in Update mode.

- o **S (Skip)** - Skips the data field (i.e., the cursor does not move to the data field in Verify mode).
- o **K (Key)** - Requires that an operator key the data. Typically, this is *blind* verification, meaning that the verify operator cannot see what the entry operator has keyed. As an entry operator types each character into the data field, it is compared to the existing character. If there is a discrepancy, the verify operator receives an error message.
- o **E (Enter)** - Allows new data to be entered into this data field.
- o **C (Acc not 0)** - If the accumulator for this field is not equal to zero, the cursor stops on it.
- o **V (Visual)** - Requires visual verification.

- o **R (Release)** - Releases the current record and moves the cursor to the first update field in the next record.

18. In the **Export position** field, select the number that determines placement of the field data in an export file. Select the number of the character position in the **Export position** field for all fields that you need to export with this method. To avoid overwriting data, be sure to separate the export position values with enough space to accommodate the field's data length. The default value is 0.

19. Select or deselect the following options:

- o **Re-verify** - If the operator makes a change to data during the Verify mode of Completion, the data needs to be entered again or re-verified before the cursor moves to the next field.
- o **Zero Fill If Data** - If the field is partially filled with data, it is padded with zeroes to equal the field length. If you do not select this option, the remaining positions are filled with spaces. Zero Fill If Data is the default setting for numeric fields.
- o **Tab Field** - The cursor skips to this field when the operator presses the [Tab] key during a Completion session. The [Tab] key moves the cursor to the next field that has the Tab Field option set. This allows an operator to quickly bypass fields. By default, the [↓] key not the [Tab] key moves the cursor from field to field.
- o **Must Enter** - The operator must enter data in this field before the next field can be accessed. An error occurs if the operator tries to release the field without entering any data.
- o **Record Ascendancy** - The value of this field must be equal to or greater than the value of the same field in the previous record.
- o **Password** - This field value will display as ***** (asterisks) during Completion or on a form displayed in a browser.
- o **Field Boundary** - If the operator fills the field with data, the [Enter] key must be pressed to move to the next field.
- o **Zero Fill If No Data** - The field is filled with zeroes when no data input. If you do *not* select this option, the field is filled with spaces.
- o **Indexed Field** - Specifies the current field as an indexed field for certain export processes.
- o **Must Complete** - The operator must fill every character location in the field before accessing the next field. An error message displays if the operator tries to release the field before filling data to the last position.
- o **Form Ascendancy** - The value of this field must be equal to or greater than the value of the same field in the previous record with the same form number.
- o **Multi-line** - This option is currently not available.
- o **Right Justify** - Right-justifies the data in the field. Default setting for numeric fields.
- o **Conditional Link** - Allows the operator to link to a data form by entering a form number in the field. The field length should be the same number of digits as the largest form number in the job, and never more than two characters. The field type must be Unsigned Numeric.
- o **Non Display** - Defines the field as not visible during a Completion session.
- o **Non Keyable** - Defines the field as read-only in Completion. Edits can be written into the field, but operators cannot change that data.
- o **Text Field** – This field is currently not available. This option is intended to identify multiple, consecutive input fields as text fields, enabling a completion operator to

type continuous text that will wrap from one field to the next. The behavior of the combined text fields will mimic that of a word processor, allowing inserted text, line wrapping, etc. For example, you can define multiple fields as text fields to create an area large enough to accommodate free form comments.

Defining Selective Edits for a Field

Selective Edit options control the Accumulator, Check Digit, and Value Table properties for the current data field. To set these values for a field:

1. Select a field on the form.

2. Display the Field Properties, Selective Edits tab for the field.

See the steps in the following sections to:

- o   Determine when to execute field edits.
- o   Establish accumulator fields.
- o   Retrieve values from previous records and forms.
- o   Set range values.
- o   Create check digits.

Setting Field Edit Properties

1. In the **Field edit** field, select the value that determines when a field edit will be executed if you are using a VBA or FPL edit program to verify the data in this field. This feature applies the field edits in the VBA project attached to the job or specified on the Edits tab in the Job Properties dialog. Options are:
   - o   **Y (End)** - Calls the field edit after the field is released.
   - o   **S (Start)** - Calls the field edit before any data is entered into the field.
   - o   **B (Both)** - Calls the field edit before and after data is entered into the field.

Note: The Completion module will not begin to run field edits on the last record in a batch if an operator has backed up and then moved forward (i.e., pressed the [Pg Down] key). If you want to force a start field edit to run under these conditions, set the FSE=Y parameter in the System.ini or Job ini file .

Setting Accumulator Properties

1. In the **Acc number** field, select a number from 1 to 16 to indicate the accumulator used for the specified batch balancing method if this field is an accumulator. An accumulator is a mathematical storage area into which the value of a numeric field is added or subtracted. There are 16 accumulators available during a Completion session or per batch. When the batch is completed, all accumulator values must equal zero for the batch to be in balance.

2. In the **Bal method** field, select a balance method from the drop-down menu if this is an accumulator field. Options are:

   o **A (Addition)** – Add the value of the data field to the value in the accumulator.

   o **S (Subtraction)** – Subtract the value of the data field to value in the accumulator.

Setting Automatic Record Properties

1. To retrieve data from previous records, in the **Record auto** field, select the function of this field during Completion when the Auto key [F2] is enabled. Options are:

   o **Duplicate** - Duplicates the field data from the same field on the previous record.

   o **Skip** - Skips the current data field.

   o **Increment** - Increments the field value by one from the same field value on the previous record.

   o **Double** - Forces the operator to double-key any data typed in the data field.

   o **Constant** - Places the constant value (specified in the **Type Mask** field) into the field data. This function only occurs on the location field during a record insert.

2. Select the **Confirm** check box to force the operator to confirm the Record Auto functions for this field during Completion.

3. To retrieve data from the form level, in the **Form auto** field, select the function of this field when the Auto key [F2] is enabled. Options are:

   o **Duplicate** – Duplicates data from the same field on the last record with the same form type as the current form.

   o **Increment** – Increments the field value by one from the same field in the last record that had the same form number.

Setting Range Options

1. To define a range of characters for the field, select the range type in the **Type** field. The type determines what data is accepted in this field during Completion. Options are:

   o **In** – Data is inside the range specified in the **Lower** and **Upper** fields.

   o **Out** – Data is outside the range specified in the **Lower** and **Upper** fields.

   o **Sel** – Data matches the specific characters in the **Lower** field. Overflow of these characters can be entered in the **Upper** field.

2. In the **Lower** field, type the lower limit of the range.

---

3. In the **Upper** field, type the upper limit of the range.

4. In the **Include** field, type the characters, or ranges of characters that are valid entries for this data field. If you allow a minus sign, it must be the first character. If you allow spaces, the space must be first (if no minus sign) or second (if the minus is allowed). You can enter all other characters in any order. For example, if you type "– A– Z12", then minus signs, spaces, uppercase letters A through Z, and the numbers one (1) and two (2) are allowed. All other characters are invalid for this field. If you do not type values in this field, all Field Type characters are included for the field, minus any characters in the **Exclude** field.

5. In the **Exclude** field, type the characters, or ranges of characters, that cannot be entered in this data field. Specify the values as you did for the **Include** field. If you want to exclude spaces, you must follow the space by another invalid character.

Creating Check Digits

You use check digits to verify the validity of the information entered into the data field. A check digit is the last digit of a field or the last digit of the final field in a multiple-field calculation. A calculation is performed on all of the numbers entered into the data field or into several data fields during a Completion session. The result of this calculation is the check digit.

You can create up to 31 check-digit algorithms. For a check digit to be calculated, the data field must be right-justified field or completely filled with data. The total number of characters in the field cannot exceed 14. Check digits are either generated or validated during the Completion process. The method chosen depends on how you use the check digit.

1. In the **Number** field, select the number of the check digit algorithm to be used on this field. Valid numbers are 1-31.

2. Select the **Multifield** check box to indicate that multiple fields will be used in this check digit calculation.

3. In the **Method** field, select check digit method for this field. Options are:

   o **Validate** - The algorithm validates the last digit entered in the field, and is executed upon release of the field. The result of the algorithm is compared to the check digit inserted into the field by the operator or by the recognition engine. If there is a discrepancy, the operator is prompted to change the data or override the check digit. The numeric field and its check digit must exist on the data form.

   o **Generate** - The algorithm generates a check digit in the last position of the field. Use this method if the check digit is not printed on the paper form, but is required for other processes in an imaging system, such as a database lookup.

4. Select the **Edit** button to display the Check Digit Properties dialog. If the check digit does not exist, a new one is created.

5. In the **Name** field, type a number from 1 to 31 to specify the check digit algorithm.

6. In the **Modulus** field, select a number that is divided into either the weighted sum of digits or the weighted sum of products (depending on the algorithm type) to calculate the check digit. The modulus can be a number from 1 to 11.

7. In the **Weight** field, type the series of numbers used to calculate the check digit.

Note: For each check digit you define, you must type a value in the **Weight** field—every check digit must have a weighting factor.

8. In the **Remainder** field, select a number between 00 and 11 from which the remainder can be subtracted to find the check digit. If the remainder is greater than the constant remainder, the check digit is the unsigned value of the calculation. If the check digit is complemented, subtract the remainder from the constant remainder and subtract the answer from the modulus. The check digit is always an unsigned value. In most check-digit routines, the constant remainder is 00.

9. Select the **Complemented** check box if the value is the difference between the modulus and the remainder.

10. Select the **Replace 10** check box if you specified 10 as the modulus. When you specify 10 as the modulus to calculate the check digit, the value of the check digit can be 10. Because the numeric field for a check digit is only one position, a double-digit check digit needs to be replaced with a single digit. Type the replacement digit, which can be any keyboard character, in the empty field next to the check box.

11. Select the **Replace 11** check box if you specified 11 as the modulus. When you specify 11 as the modulus to calculate the check digit, the value of the check digit can be 10 or 11. Because the numeric field for a check digit is only one position, a double-digit check digit needs to be replaced with a single digit. Type the replacement digit, which can be any keyboard character, in the empty field next to the check box.

12. Select the **Override** check box if the value that you type in the empty field next to this check box will override the check digit.

13. In the **Type** area, select the type of check digit algorithm you are defining. Types include:
    o **Weighted Sum of Digits -** Generate or validate a check digit. The weighting factor is multiplied against each digit of the base number, resulting in a product for each number that was entered in the field. The digits that make up each product are then added together and the modulus divides this sum. The remainder is the check digit. You can also complement the check digit by subtracting it from the modulus.
    o **Weighted Sum of Products** – Generate or validate a check digit. The weighting factor is multiplied against each digit of the base number, resulting in a product for each number that was entered in the field. Then all products are added together and the modulus divides this sum. The remainder is the check digit. You can also complement the check digit by subtracting the check digit from the modulus.
    o **Division of Entire Number** – Validate a check digit only. If you select Generate, Division is ignored. The entire number (the base number and the check digit) entered in the field is divided by the modulus. The remainder is the check digit. You can also complement the check digit by subtracting it from the modulus. The weighting factor does not apply.
    o **Division of Base Number** - Generate or validate a check digit. The base number (excluding any check digit) is divided by the modulus. The result of this division is the check digit. You can also complement the check digit by subtracting it from the modulus. The weighting factor does not apply.
    o **Geometric, Base 2** - Generate or validate a check digit. With this check digit, 2 to the power of the weighting factor, multiply each digit of the base number. This results in a product for each number entered in the field. All products are then added and the modulus divides this number. The remainder is the check digit. You can also complement the check digit by subtracting it from the modulus.

14. Select **OK** to save the check digit algorithm properties and return to the Selective Edits tab.

Weighted Sum of Digits Example

- The operator types the number "98765," the base number.
- The modulus is 10.
- The check digit is not complemented.
- The weights are 65432.

1. Determine the product of each digit in the field multiplied by the weight. The weights are applied to the base number from right to left and repeated until each number has a weighting factor.

```
Number entered   9    8    7    6    5
Weights          2    3    4    5    6
                _____
Products        18   24   28   30   30
```

2. Add the digits of the products together.

```
Sum of Digits: 1 + 8 + 2 + 4 + 2 + 8 + 3 + 0 + 3 + 0 = 31
```

3. Divide the weighted sum of the digits by the modulus.

```
31/10 = 3 r1
```

4. Because the digit is not complemented, the check digit is 1. Therefore, the check digit of 98765 is 1.

Weighted Sum of Products Example

- The operator types the base number "1853954."
- The modulus is 11.
- The check digit is complemented.
- The weights are 123.

1. Determine the product of each digit of the base number multiplied by a weighting factor.

```
Number entered   1    8    5    3    9    5    4
Weights          1    3    2    1    3    2    1
                _____
Products         1   24   10    3   27   10    4
```

2. Add the sums created in step 1.

```
Weighted Sum of Products 1 + 24 + 10 + 3 + 27 + 10 + 4 = 79
```

3. Divide the weighted sum of products by the modulus.

```
79/11 = 7 r2
```

4. Because the check digit is complemented, subtract the remainder from the modulus to find the check digit.

```
11 - 2 = 9
```

5. The check digit for the number 1853954 is 9.

Division of Entire Number Example

This method can be used only for validation.

- The operator types the number "38657." The entire number is "3865" while the check digit is 7.

- The modulus is 9.
- The check digit is complemented.
- There are no weights with this check-digit method.

1. The number typed into the field (the base number) is divided by the modulus.

   ```
   38657/9=4295 r 2
   ```

2. Because this digit is complemented, the remainder is subtracted from the modulus.

   ```
   9 - 2 = 7
   ```

3. The last number entered by the operator was a "7," and the calculation resulted in a "7." This is a valid check digit.

Division of Base Number Example

- The operator types the base number "1256."
- The modulus is 10.
- The check digit is complemented.
- There are no weights with this check-digit method.

1. The base number is divided by the modulus.

   ```
   1256/10=125 r 6
   ```

2. The remainder of the above calculation is "6." Since this is a complemented check digit, the remainder is subtracted from the modulus.

   ```
   10 - 6 = 4
   ```

3. This calculation yields the check digit. The check digit for 1256 is 4.

Geometric Base 2 Example

- The operator types the base number "3489."
- The modulus is 10.
- The check digit is not complemented.
- The weights are 23.

1. Determine the product of each digit of the base number multiplied by a weighting factor. The weights are applied to the base number from right to left and repeated until each number has a weighting factor.

   ```
   Base Number              3    8    9
   Weighting Factors       2³   2²   2³   2²
                          _____
   Products                24   16   64   36
   ```

2. Add the sums created in step 1.

   ```
   Weighted Sum of Products 24 + 16 + 64 + 36 = 140
   ```

3. Divide the weighted sum of products by the modulus.

   ```
   140/10 = 14 r 0
   ```

4. Because the check digit is not complemented, do not subtract the remainder from the modulus.

5. The check digit for the number 3489 is 0.

Setting Value Table Properties

A value table is a lookup table that validates the data entered in the field. A value table can be a file or an index that points to a file. If you assign a value table to a field, the table lookup is performed whenever an operator types data into the specified field and presses the [Enter] key. All comparisons are performed on field one of the value table file.

You create value tables using the Administration module. HOW????? Value tables are named in a standard format, VALTBL##.VTB, where ## represents a two-digit numeric value from 01 to 99. You use this number on the Selective Edits tab to reference the table.

To create a value table and assign it to a field:

1. Open FormWare Administration.

2. In the File Browser, select the Value Table folder.

3. Right-click in the Contents area.

4. Select **New** from the context menu. The Value Table Properties dialog displays.

5. In the **Name** field, type the two-digit number that identifies this value table.

6. In the **File Name** field, type a valid batch to be used as a value table. When you assign a path and file name here, you can use subdirectories. The syntax is \<subdirectory name>\<file name>. The software automatically adds the <drive letter>:\FormWare\Files.

7. Select the **Indexed** check box if the batch is indexed.

8. Select **OK** to create the value table.

Note: The batch and index must be in the appropriate folder.

9. Open the job and form in Designer.

10. Double-click the field for which you want to assign a value table to display the Field Properties dialog.

11. Select the Selective Edits tab.

12. In the **Method** field in the Value Table area, select from the following values:

   o **Validate** – Compares the data entered in the data field to the specified value table to determine if data is valid. If the data does not match an entry in the value table, the operator receives an error message.

   o **Display** – Searches the value table for the data entered in the data field during Completion. Depending on the data in the field, additional information is displayed on the Completion status line.

   o **Replace** – Searches the value table for the data entered in the data field. Information in the value table fields is inserted into subsequent fields of the current record. The fields filled with data from the value table are then skipped. The fields into which data is placed and the fields in the value table must be the same length.

Note: When you add a Value Table to a field on the Selective Edits page, you should select the method you are going to use before accessing the Value Table dialog. Method options are: **Validate**, **Display**, or **Replace**.

13. In the **Number** field in the Value Table area, select the number of the value table to be used for this field. Valid numbers are 1–99.

> **Note**: If you have not yet defined a value table, select the **Edit** button to display the Value Table Properties dialog and create one.

14. Select the **Invalid Entries** check box, if the value table for this field contains a list of invalid, rather than valid, entries.

15. Select **Save All** to save your settings.

Adding Rule Files

In the **Rule Name** field, select the **Browse** button to locate the Rule file for this field.

To remove the rule, select the **Delete** button.

Creating Field Input Masks

In the **Input Mask** field, type formatting information provided to the Completion operator when entering data. Underscores represent the characters typed by the operator. Formatting characters are for display only and are not added into the record. For example, if the input mask is (_ _ _)_ _ _–_ _ _ _ and the operator types 4356557350, the information is displayed as (435) 655–7350. The input mask is ignored if each character in the field is not represented by an underscore.

Creating Field Type Masks

In the **Type Mask** field, type the character type of each position within the data field. If you type values in this field, they are used for both pattern matching during the Recognition process and as a type mask during Completion.

Understanding Type Masks

For example, some account numbers and postal codes require an alphabetic character in the first position and numeric characters in other positions. The type mask allows you to specify the types of the individual characters, not just the field. For example, a type mask of AAANN would allow alphabetic data in positions one through three and numeric values only in positions four and five. Each position in the field must be represented by a value in the type mask, otherwise the type mask is ignored in Completion. For example, a field with a length of 5 would also have to have a type mask of the same length. Valid types are:

- o **A** – Alphabetic character
- o **N** – Numeric character
- o **L** – Alphabetic or numeric character

Understanding Recognition Pattern Matching

Recognition pattern matching is similar to, although less powerful than, standard regular expression algorithms. The values you type in the **Type Mask** field define the recognition pattern matching string, that is, which character type should be in a certain position within the field's recognition zone.

The pattern string describes both exactly how many characters there should be and what each character should be. To avoid matching possible false positive characters in the first position of the recognition zone, a pattern "scrolling" function is implemented during the pattern matching process. If the number of recognized characters for the zone is greater than the pattern length, the pattern will be matched up with the zone recognition results starting with the first character of the

zone. If the matching pattern is not found, the pattern search is repeated starting with the second character of the zone, etc. The first string of data that matches the pattern is extracted for the field.

During the search for matching characters, alternate guesses for each character are also analyzed. If the first guess is not a match, and one of the alternate characters is a match, the alternate character will be promoted to the first guess. The original first guess character will be swapped with the new first guess as an alternate. If no exact match is found, the results of the first match (started from the first character) will be used.

When any character within the zone recognition results does not match the pattern, its confidence level is lowered to 1. If the number of characters in the zone (n) is fewer than specified by the pattern, the first n characters in the pattern are applied. If the number of recognized characters is greater than the pattern string length, then the confidences of all of the characters lying outside of the match string are lowered to 1. (1 is the lowest legal confidence value allowed within recognition.) If the number of recognized characters is less than the pattern string length, then the confidences of all the characters in the recognized string are lowered to 1 (no match).

Note: The pattern-matching algorithm will not reject the field under any circumstances.

The following values are valid to use for pattern matching within the **Type Mask** field:

A – Any alphabetic character

N – Any numeric character

L – Any character

{ – Start of enumeration

} – End of enumeration

Here are some examples of the type masks for pattern matching:

AANNNAA – First two characters are alphas, followed by three digits, followed by two alphas.

NNN{-}NNN – Three numbers followed by a hyphen followed by three numbers.

A{1234}NN – An alpha followed by a single number in the range from one to four followed by two numbers.

If you plan to use pattern-matching functionality it is important to note the following:

Certain settings on the Field Properties, Selective Edits tab may impact the pattern-matching algorithm. For example, you have defined a field with the following characteristics:

- A recognition zone of Alphanumeric
- **Include** field within the Range area is defined as A-Z.
- The value in the **Type Mask** field is defined as AANAAA.

In this case, the third character "N" will never be matched (output will be confidence = 1), because the **Include** field range prevents numeric characters from being used for recognition results.

Recognition pattern matching occurs as a *post-recognition processing function* and applies to all recognition engines available within FormWare. However, any internal recognition engine algorithm that may be applied may be quite different and could produce unexpected results.

The order of processing within recognition will be as follows:

1. Recognition

2. Free Form processing

3. Table processing

4. Pattern matching

5. Automated Data Edits (ADEs)

Adding a Bitmap Field

You can add a JPG image to a data form to display during Completion. This is typically used for I-Forms. To add a bitmap image:

1. From Designer, select the **Bitmap** button from the Field Types toolbar and click the spot on the data form where you want the bitmap to display.

2. Double-click the field if the Field Properties dialog is not displayed.

3. Select the Image tab.

4. In the Bitmap area, select the **Browse** button for the **File** field. The Browse Files dialog displays the contents of the \FormWare\Bitmaps folder.

5. Select the bitmap image file you want to place on the data form and select **Open**. The image must be a JPEG file. The image displays on the data form.

6. For the **Display Mode**, select **Opaque** to display the image in its entirety, blocking anything on the form behind it. Select **Transparent** to display the image with all colors visible except the color you select in the drop-down list. The color you select will drop out of the image causing those areas in the image to become transparent (i.e., any place the selected color existed on the image will now show the form underneath).

7. For the **Effect**, select the placement and display of the image. Options are:
    o **Center** – Centers the image in the middle of the field.
    o **Stretch** – Expands the image to fill the entire field. This effect can cause the image to become distorted.
    o **Tile** – Replicates the image, while retaining the original size, as many times as needed to fill the entire field.
    o **Actual** – Displays the image as it is imported.

8. Select the **Save All** button on the Designer window toolbar.

Editing Fields and Labels

Cutting, Copying, Pasting, and Deleting Fields

Once you create fields and labels, use the standard Windows functions to cut, copy, paste, and delete those items on your data form. Except for the **Delete** function, these functions also display as buttons on the Designer window toolbar, on the Designer window Edit menu, and on the data form context menu.

The following instructions indicate how to cut, copy, paste, and delete using the data form context menu.

1. Select one or more fields/labels on the data form by clicking on them with your mouse. To select multiple items, hold down the [Shift] key while you select the objects with your mouse or click and drag your mouse over the data form to "rubber band" a group of items.

2. With the appropriate fields/labels selected, right-click to display the data form context menu.

3. Select the option to perform the function you want. For example:

    a. Select **Cut,** then **Paste** to move one or more fields or labels to a new location on the data form.

    b. Select **Copy,** then **Paste** to copy one or more fields or labels and paste them in a new location on the data form.

Note: When you copy fields and labels, you copy all the properties that have been defined for that field or label.

    c. Select **Delete** to permanently remove one or more fields or labels on the data form.

    d. Select **Delete Image Zone** to remove a specified image zone from the image displayed in the viewer. This option only works on one zone at a time.

4. Select **Save All** on the Designer window to save your changes.

Changing a Field Name

To quickly change the name of an existing field within a job:

1. Highlight the name of an existing field within the Designer window Job Browser.

2. Right-click to display the context menu and select the **Rename** option.

3. Type the new field name. This name must be unique among the other field names on the form. You may want to refer to the field in a rule so make sure that the name you use is identifiable. Valid characters for a field name are A-Z, a-z, 0-9, _, and -. Any other character is unacceptable and will display an error message.

4. Select **Enter.**

5. Select the **Save All** button on the Designer window toolbar to save your changes.

Editing Properties for Multiple Fields

One of the most time-consuming tasks within form design is editing the properties of existing fields and labels on the form. Use the Multi-Field Properties dialog to simplify and expedite this process by changing a field property once for multiple fields or labels.

The "Select All" options on the Designer window Edit menu enables you to globally select all fields, labels, or fields and labels on the form with a single keystroke. You can use this feature to select multiple fields or labels on the form and then change the properties of those fields or labels without having to edit the properties for each field or label individually.

Understanding Multi-Field Properties

When you select two or more fields on the data form and select Properties from the context menu, the Multi-Field Properties dialog displays. Use this dialog to globally change field-level properties for all fields and labels currently selected on the data form.

There are three pages on the Multi-Field Properties dialog: General, Edits, and Recognition. Each page contains some of the properties that are available for a single field on the corresponding Field Properties dialogs within Designer. Properties were chosen for the Multi-Field Properties dialog based on the necessity to change them for more than one field at a time.

Based on the way that fields on this dialog function, it is important to understand the following:

When you modify any property on the Multi Field Properties dialog (i.e., "dirty it") it is added to a list of properties to apply to the selected fields. These properties are automatically applied to the selected fields when you:

- Deselect that group of fields by selecting the form
- Select a new field
- Do something to close the form editor

Properties are only applied if they have been changed on the Multi-Field Properties dialog. There may be situations when you must select/change a property and then deselect/change it to arrive at the behavior you desire.

For example, say that out of a group of six selected fields, three of them have the Field boundary property checked. To uncheck (remove) this property for all six fields you must to do the following:

- First, select all six fields, display the Multi-Field Properties dialog for those fields, check the Field boundary property to dirty it, and exit the dialog.
- Then, select all six fields again, display the Multi-Field Properties dialog for those fields, and uncheck the Field boundary property it to arrive at the setting you desire.

Accessing the Multi-Field Properties Pages

There are a number of ways to access the Multi-Field Properties dialog for multiple fields and labels on the form.

To access the Multi-Field Properties dialog for a select number of fields and labels:

1. Activate the data form on the Designer window that contains existing fields or labels.

2. Hold down the **[Shift]** key and click the fields and labels you want to select with your mouse.

3. If you do not have the properties dialog push pin set, right-click to display the data form context menu and select **Properties** to display the Multi-Field Properties dialog.

To access the Multi-Field Properties dialog for all fields and labels:

4. Activate the data form on the Designer window that contains existing fields or labels.

5. Select the Edit menu and then one of the following options:
    o   **Select All (Ctrl+A)** – Selects all fields and labels on the data form.
    o   **Select All Data Fields (Ctrl+D)** – Selects all fields on the data form (omitting labels).
    o   **Select All Labels (Ctrl+L)** – Selects all labels on the data form (omitting data fields).

6. A dialog displays asking if you want to include non-display fields in your field selection. Select **Yes** to include or **No** to exclude non-display fields on the form from the group of fields you are selecting.

Setting General Properties for Multiple Fields

The options on the Multi-Field Properties, General page enable you to modify some of the basic field properties for a group of selected fields. On this page, you can change field type, size, display mode, font, background color, and text color of multiple fields and labels. These options are identical to the options you set for a single field on the Field Properties, General page, but here you are setting these properties for multiple fields.

To change the general field properties for a group of fields or labels:

1. On the Multi-Field Properties, General page, define the following for all selected fields:
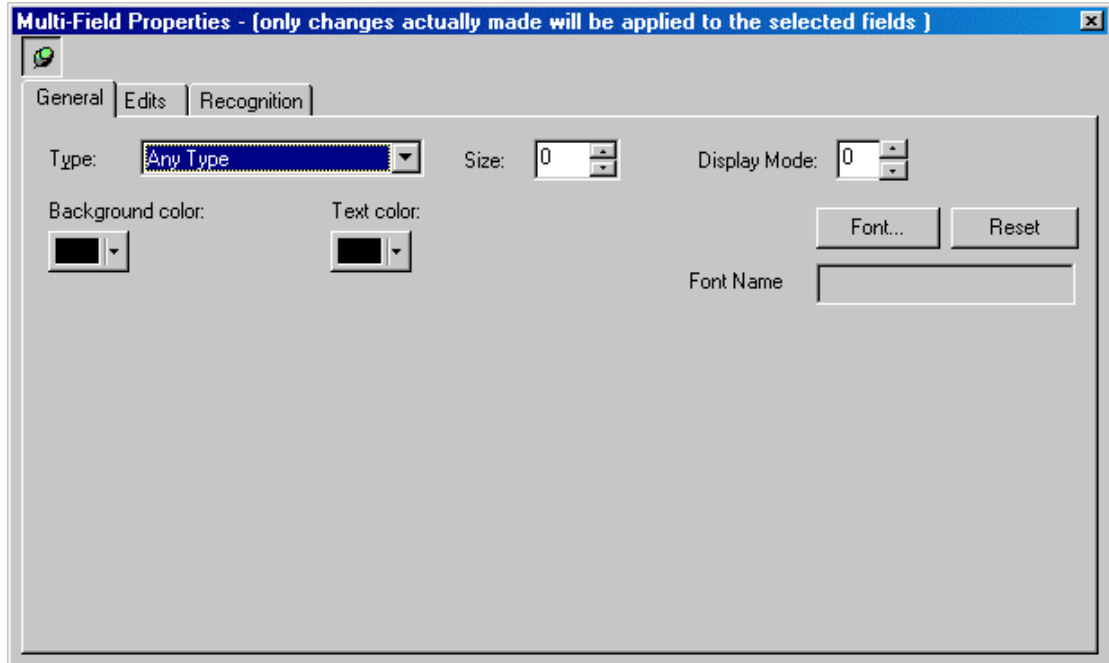
2. In the **Type** field, select the field data type for this group of fields. Options are:
    o   **Any Type** – A data field that accepts alphanumeric characters. This includes the entire ASCII character set except the backslash (\), which acts as a field release. Select Any Type if you want to apply free form recognition to this field.
    o   **Alpha Only** – A data field that accepts only alphabetic characters.
    o   **Upper Case** – A data field that accepts only upper case alphabetic or numeric characters, A–Z, 0–9.
    o   **Unsigned Numeric** – A data field that accepts only numeric characters 0–9 but not any sign characters, such as a minus sign.
    o   **Signed Numeric** – A data field that accepts only numeric characters, 0–9 and the minus sign.
    o   **Mostly Numeric** – A data field that accepts alphanumeric characters, including sign characters. Use this type for fields such as Social Security Numbers, which are mostly numeric.

3. In the **Size** field, type or use the scroll arrows to change the size of all selected fields by number of characters.

4. In the **Display Mode** field, type the number that corresponds to the display mode for all selected fields within a multi-pass application.

---

5. In the **Background Color** field, select the background color to display for all selected fields. The drop down arrow displays the standard Windows color palette.

6. In the **Text Color** field, select the text color to display for all selected fields. The color you select here overrides color settings made on the Font dialog and on the System Defaults dialog. The drop down arrow displays the standard Windows color palette.

7. In the **Font Name** field, change the font currently selected for the selected labels.

    a. Press the **Font** button to display the Font dialog, where you can select a different font.

    b. Press the **Reset** button to reset the Font selection to the system default.

8. Select **Save All** on the Designer window toolbar to save your changes.

Setting Edit Properties for Multiple Fields

The options on the Multi-Field Properties, Edits page enable you to modify some of the selective and check box edit field properties for a group of selected fields. There are two sections on this page, one for selective edits and one for check box edits. These options are identical to the options you set for a single field on the Field Properties, Checkbox Edits and Selective Edits pages, but here you are setting these properties for multiple fields.

To change the edit field properties for a group of fields or labels:

1. On the Multi-Field Properties, Edits page, define the following for all selected fields:
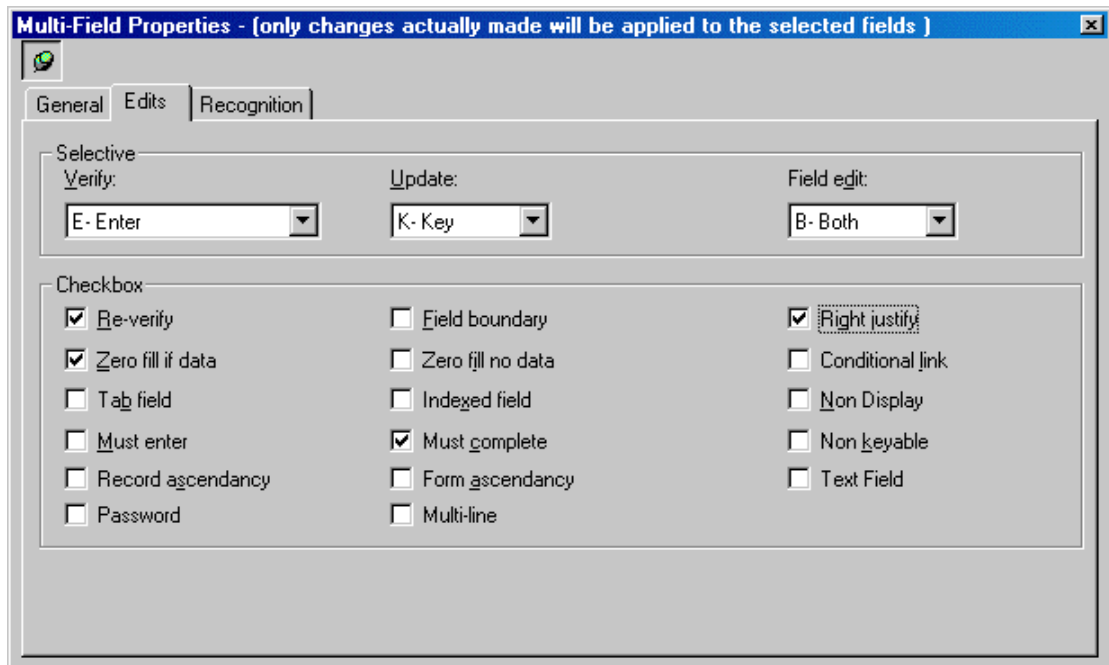
2. In the **Verify** field, select the value that sets the behavior for the selected data fields when the Completion module runs in Verify mode. Options from the drop down list are:

    o **S-Skip** – Skips the data field (i.e., the cursor does not move to the data field in Verify mode).

    o **K-Key** – Requires that an entry operator key the data. Typically, this is blind verification, meaning that the verify operator cannot see what the entry operator has keyed. As an entry operator types each character into the data field, it is compared to

the existing character. If there is a discrepancy, the verify operator receives an error message.

- o **E-Enter** – The cursor stops on this field for every record and allows the entry of data in a previously blank data field.
- o **C-Acc not 0** – If the accumulator for this data field is not equal to zero, the cursor stops on it to be key-verified.
- o **V-Visual** – Requires visual verification. If you want to use the parameter QCO=Y (Quality Check rejects only), you must set the field type to Visual verify.
- o **R-Release** – Ignores all other Verify options and releases the record.

3. In the **Update** field, select the value that sets the behavior for the selected data fields when the Completion module runs in Update mode. Options from the drop down list are:

- o **S-Skip** – Skips the data field (i.e., the cursor does not move to the data field in Update mode).
- o **K-Key** – Requires that an entry operator key the data. Typically, this is blind update, meaning that the update operator cannot see what the entry operator has keyed. As an entry operator types each character into the data field, it is compared to the existing character. If there is a discrepancy, the update operator receives an error message.
- o **E-Enter** – Allows new data to be entered into this data field.
- o **C-Acc not 0** – If the accumulator for this field is not equal to zero, the cursor stops on it.
- o **V-Visual** – Requires visual verification.
- o **R-Release** – Releases the current record and moves the cursor to the first update field in the next record.

4. In the **Field Edit** field, select the value that sets the behavior for an edit program written in VBA or FPL to the selected data fields. When you attach a VBA project to the job, by either creating a VBP with the same name as the job or manually on the Job Properties, Edits page, field edits within that program are applied to the selected fields during the Completion process. The option you specify here determines when the edit program is applied to the field(s). Options from the drop down list are:

- o **Y-End** – Calls the field edit after the field is released.
- o **S-Start** – Calls the field edit before any data is entered into the field.
- o **B-Both** – Calls the field edit before and after data is entered into the field.

Note: The Completion module will not run field edits on the last record in a batch if an operator has backed up and then moved forward (i.e., pressed the [Pg Down] key). If you want to force a start field edit to run under these conditions, set the FSE=Y parameter in the System.ini file (for all jobs) or job INI file (for a specific job).

5. Select the **Re-verify** check box if you want to force the operator to re-key or re-verify data in these fields again during Completion, Verify mode before proceeding to the next field on the form.

6. Select the **Field boundary** check box to enable the field boundary for the selected fields. If operators fill the selected fields with data, it will require them to press [Enter] to advance to the next field on the form.

7. Select the **Right justify** check box to right-justify data in the selected fields. This is the default setting for numeric fields.

8. Select the **Zero fill if data** check box to pad the selected fields with zeroes if those fields are partially filled with data. If you do not check this option, the remaining positions are filled with spaces. This option is the default setting for numeric fields.

9. Select the **Zero fill no data** check box to fill the selected fields with zeroes when no data is input. If you do not check this option, the selected fields are filled with spaces.

10. Select the **Conditional link** check box to link the selected fields to a data form by typing a form number in the fields. Set the field length for the selected fields to accept the maximum number of digits of the largest form number within the job. The field length should never be set to more than two characters (to accept a two-digit form number). The field type for the selected fields must be Unsigned Numeric.

11. Select the **Tab** field check box to enable the operator to access the selected fields using the [Tab] key during a Completion session. The [Tab] key moves the cursor to the next field that has the Tab Field option set. This enables operators to quickly bypass certain fields on the form.

Note: By default, the [Tab] key does not move the cursor from field to field, the [ ] key does.

---

12. Select the **Indexed** field check box to identify the selected fields as indexed fields for certain export processes.

13. Select the **Non Display** check box to hide the selected fields from the operator during a Completion session.

14. Select the **Must enter** check box to force the operator to enter data in the selected fields before advancing to the next field on the form. An error occurs if the operator tries to move past the selected fields without typing any data.

15. Select the **Must complete** check box to force the operator to fill every character position within the selected fields before advancing to the field on the form. An error message occurs if the operator tries to move past the selected fields before filling all character positions with data.

16. Select the **Non keyable** check box to identify the selected fields as read-only in Completion. Edits can be written into fields of this type, but operators cannot change that data.

17. Select the **Record ascendancy** check box to force the value of the selected fields to be equal to or greater than the value of the same field in the previous record.

18. Select the **Form ascendancy** check box to force the value of the selected fields to be equal to or greater than the value of the same field in the previous record with the same form number.

19. Select the **Text Field** check box to identify multiple, consecutive input fields as text fields, enabling a Completion operator to type continuous text that will wrap from one field to the next. The behavior of the combined text fields will mimic that of a word processor, allowing inserted text, line wrapping, etc. For example, you can define multiple fields as text fields to create an area large enough to accommodate free form comments.

20. Select the **Password** check box to force the display of a password value typed into the selected fields within Completion or on a form as ***** (asterisks). This protects the password value by masking the display of the actual characters of the password.

21. Select the **Multi-line** check box to indicate that the selected fields use multiple lines on the form.

22. Select **Save All** on the Designer window toolbar to save your changes.

Setting Recognition Properties for Multiple Fields

The options on the Multi-Field Properties, Recognition page enable you to modify some of the recognition field properties for a group of selected fields. These options are identical to the options you set for a single field on the Field Properties, Recognition page, but here you are setting these properties for multiple fields.

To change the recognition field properties for a group of fields or labels:

1. On the Multi-Field Properties, Recognition tab, define the following for all selected fields:
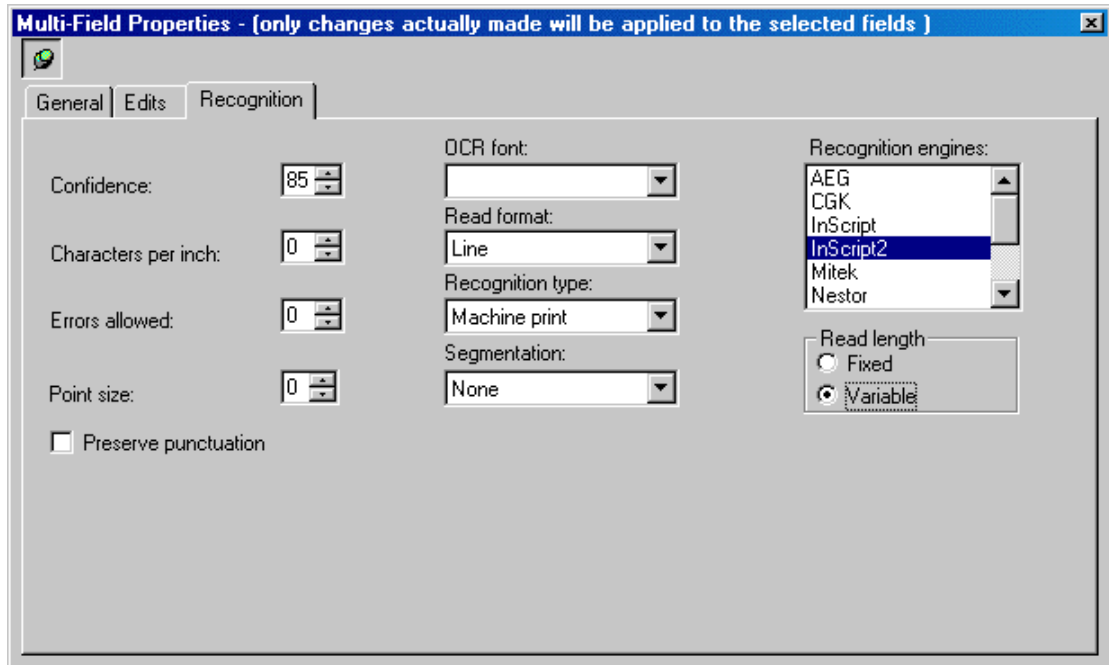
2. In the **Confidence** field, set the value that indicates how confident the recognition engine must to be to accept characters for the selected fields. This value is represented as a numeric percentage from 0 to 100%. You can type a value or use the arrow keys to raise or lower the current value. All characters with a confidence level value below this number will display as character rejects during a Completion session.

3. In the **Characters Per Inch** field, set the value that indicates the number of characters per inch on the original document for the selected fields. This value is represented as a number from 0 to 100. You can type a value or use the arrow keys to raise or lower this value.

   Setting this option helps the recognition engine segment the characters in the field correctly and can improve recognition. To determine the CPI, measure a one-inch segment on an original form and determine how many characters are located in the inch. For example, if your recognition zone is two inches long and your CPI is 5, the recognition engine would slice the zone into ten equal sections and then try to recognize a character in each zone. This setting applies to all the fields on your form.

   You can also select particular fields on the Data Form and change the CPI setting for the selected fields only. Click and drag your mouse to highlight the fields you want to change. With the selected fields highlighted, right-click to display the Context menu. Select the Select Chars Per Inch option. Change the setting in the CPI dialog and click OK to continue. This setting change applies only to the highlighted fields.

Note: This property is only available with the Nestor, AEG, and CGK recognition engines (see the Recognition engines field).

---

4. In the **Errors Allowed** field, set the value that indicates the maximum number of low confidence characters permitted in the selected fields before the field is rejected during Completion. For example, if you set a value of 5, and 7 characters are rejected, all of the values in the field will be rejected. If 4 characters are rejected, the individual reject characters are displayed in Completion but the field is not rejected.

5. In the **Point Size** field, set the value that indicates the minimum point size of characters for the selected fields. Any smaller characters will be ignored during recognition.

6. In the **Preserve Punctuation** field, check the box to indicate whether or not to preserve punctuation in output data (question marks, periods, and hyphens) for the selected fields. If you do not check this option, these characters are deleted from output data.

7. In the **OCR font** field, select the option that determines which trained font to apply to the selected fields. Options are:
   o **REG**
   o **OCRA**
   o **OCRB**
   o **MICR**
   o **Farrington**

Note: The Nestor Reader engine supports OCRA, OCRB, Farrington, and MICR. You must purchase the Nestor trained fonts separately and they can only be used on numeric fields. The AEG engine supports OCRA for alpha and numeric data, OCRB and the Farrington fonts for numeric data, and MICR.

---

8. In the **Read format** field, select the option that determines how data in the selected fields are treated. Options are:
   o **Word** – Only single words without spaces are recognized. If spaces are encountered in the fields, they are compressed out of the field's data string.
   o **Line** – Permits single lines of text (with spaces) to be recognized. If spaces are encountered, they are preserved in the field's data string.Paragraph – Permits multiple lines of text (with spaces) to be recognized on the image with a single recognition zone. Lines within the paragraph are auto-detected and parsed into the appropriate data field. In the Recognition type field, select the option that determines the general type of data to recognize for the selected fields. Options are:
   o **None** – No recognition is performed for the selected fields.
   o **Machine Print** – Recognizes machine-printed alphanumeric characters that have consistent, predictable shapes for the selected fields.
   o **Hand Print** – Recognizes alphanumeric characters that vary in shape, such as hand-printed characters for the selected fields.
   o **Mark Sense** – Recognizes checkmarks, Xs, or other marks placed in check boxes for the selected fields.
   o **Bar Code** – Recognizes industry-standard bar codes for the selected fields. Bar codes are self-contained information that is encoded in the widths of printed bars and spaces.
   o **Automatic** – Attempts to automatically determine the type of characters to be recognized for the selected fields.

---

   o   **Cursive** – Recognizes cursive handwritten characters for the selected fields.

Note: Not all recognition engines support all recognition types.

9. In the **Segmentation** field, select the option that determines the amount of spacing between characters for the selected fields. Options are:

   o   **None**

   o   **1 (minimum)**

   o   **2**

   o   **3 (average)**

   o   **4**

   o   **5 (maximum)**

Segmentation provides the recognition engine with an estimate of how the characters are spaced, which enables the engine to determine where each character should begin and end. If you have characters that are very wide, set the segmentation to 5. If you have characters that are close together, set the segmentation to 1. If you specify none, the engine attempts to determine the segmentation size without this setting.

10. In the **Recognition engines** field, select the recognition engines to apply to the selected fields. If you have data from a recognition engine that is not listed, but is FormWare compliant, select the Other option. You can specify multiple engines to be executed on a field. However, you need to enable recognition voting if you are using multiple engines.

11. In the **Read length** field, select the option that determines how the recognition engine reads characters within the selected field. Options are:

   o   **Fixed** – Indicates that the number of characters read by the recognition engine is expected to equal the length of the field. Field data that does not equal the field length is rejected.

   o   **Variable** – Indicates that the number of characters read by the recognition engine does not need to equal the length of the field. The data can be any length less than or equal to the field length.

12. Select **Save All** on the Designer window toolbar to save your changes.

Defining Automated Data Edit Fields

During the design phase of creating a forms processing application, when you are defining the fields on your forms, you may determine that some of the same fields will be used repeatedly. Within Designer, you can create fields to accommodate commonly-formatted data such as phone numbers, dates, Social Security Numbers, and addresses. In FormWare, these are called Automated Data Edit (ADE) fields. Once you define an ADE field, you can apply validations that occur during Recognition, Completion, and I-Forms as data is processed to find input formats. You can also apply the same ADE field properties to fields on other forms in different jobs.

Adding an ADE Field to a Form

Some ADE fields are provided within FormWare. To use an existing ADE field:

1. Open a job and form in Designer and move your cursor to the data form area on the Designer window.

2. Right-click on the form and select **New Field > Automated Data Edit**. The Automated Data Edits dialog displays.
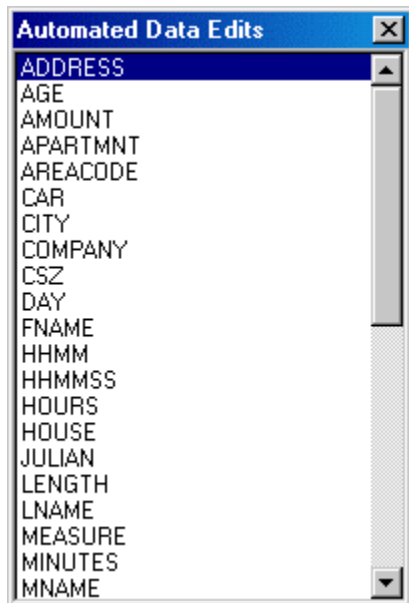
3. Double-click the field type you want in the list and then click on the data form where you want to place the field.

4. Double-click the field to display the Field Properties, General tab. The ADE type and number display in the **Edit Type** field on the General tab.

5. Select the arrow next to the **Edit Type** field. The Automated Data Edit Properties, Format tab displays.
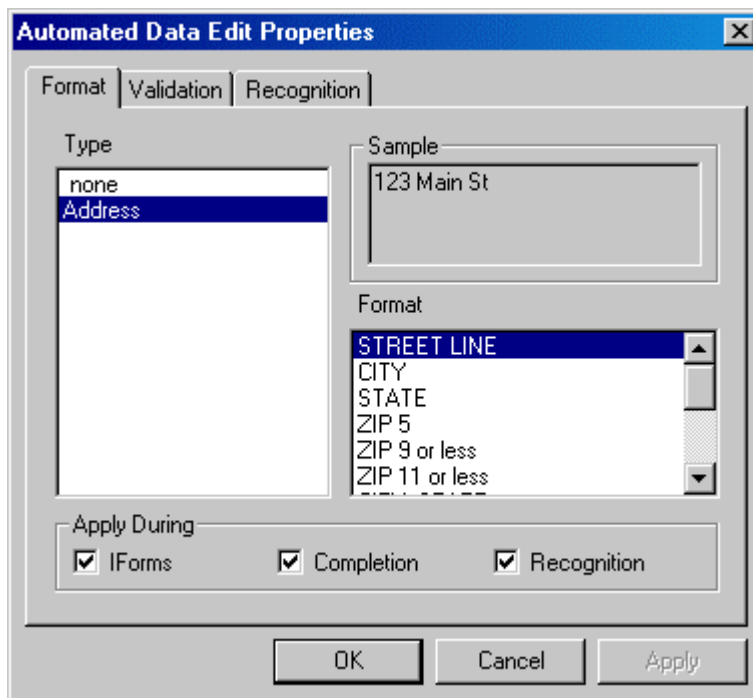
Note: You can also add a field to your form, select the arrow next to the **Edit Type** field, and then choose from a short list of ADE field types in the **Type** field.

6. In the **Format** field, select the format you prefer from the pre-defined list. You may have more than one Address field on your form with different formats: Street Line, City, State, and Zip 5. Designate each field as an Address Type field, but select different formats.

Regarding International Formats: Format types for name, address, date, amount, and phone can vary by language and by country, but within a single country (i.e., language) the **Phone ADE** cannot support both three- and four-digit exchanges. This format constraint transcends concern over translation to the equivalent word in the foreign language. For at least amount and phone ADEs, the formats may actually change MEANING too—e.g., DDDcc might change to FFFppp in French (for francs and three-digit cents). Some countries have a "NNNN" which varies; some exchanges have three-digit phone numbers; others have four-digit numbers. But, within a single country the Phone ADE can only support either 3 or 4 digit exchanges.

---

7. Select the **IForms, Completion,** and/or **Recognition** check boxes to apply the field type during these processes. By default, all three are selected.

   o **IForms** - Enforces format and validation. Under I-Forms, lookup tables require additional options to control download efficiency.

   o **Completion** - Applies ADE processing during Completion. Completion enforces format and validation with data validated upon operator exit from the field. Completion stores any modified field data results.

   o **Recognition** - Enforces format, validation, and recognition when applying ADE processing during Recognition. Data are extracted from surrounding information, validated, and then output to *OCRObject.dll* for rules and VBA processing. Validation status (success or failure) is available to rules and VBA during this process. Failed validation generates a message for statistical diagnosis, which can be viewed in reports. FormWare checks for blanks processing on all format types by copying the data and removing leading/trailing spaces that may have come from a short field in the OCR file. FormWare then uses the cleaned-up copy to decide if the field is blank and to pass to lower level validation functions.

Note: On the Format tab, the format matches the **Type** selected. The options on the Validation and Recognition tabs are also determined by the settings on the Format tab.

---

8. If this field requires validation, select the Automated Data Edit Properties, Validation tab. For example, you could use a Lookup Table to validate the State and Zip code of the Address field. If there is no input data available that conforms to the validation parameters (e.g., year must be in range 1998-2002), the program ignores the validation. For example, if the format is MMDDYY, then "Specific Ranges–Century" is ignored in I-Forms and Completion. If Recognition allows a MMDDCCYY format, then "Specific Ranges – Century" will validate the century of recognized dates.

Note Regarding Manual Validation: If you are using VBA to invoke ADE processing manually, you should generally uncheck the **Apply During** check box on the ADE Properties dialog. If extra validation processing is needed, it can usually be done after automatic ADE validation has occurred. Therefore, it would be very rare to need manual ADE Validation.

---

9. If, during Recognition, you want the engine to apply other options to the field, select the Recognition tab. For example, you could ensure that the zip code format options allow for the maximum number of digits and that both city and state names are normalized according to the table entries. The Recognition tab is a multi-select list of the formats on the data form that can be translated into the output format specified on the Format tab. The Recognition tab for Amount properties, for example, has sectors for format, separators, monetary symbols, and pad integers. By default, all input formats for Recognition are selected—i.e., data in any of these formats will be acceptable on the input form images. Select a format to deselect it.

---

10. When you have finished setting the ADE Properties, select **OK** to save the settings and close the dialog. Designer adjusts the field length to the match the format (e.g., six characters for MMDDYY), and possibly the Include/Exclude character list on the Field Properties, Selective Edits tab.

Creating User-Defined ADE Fields

If none of the pre-defined ADE fields apply to your field, you can create a custom field template. You can create fields that are unique to your company that would be used for many forms (e.g., sales region, customer number, or employee ID). To create a user-defined field:

1. Highlight the field within the Job Browser for which you want to create a user-defined ADE field.

2. Right click on the form and select **New Field > Automated Data Edit Editor**.

3. On the confirmation dialog, select **Yes** to create an ADE field type from the current field. The Automated Data Edit Name dialog displays.

4. In the **Enter ADE field name** field, type a user-defined name for your ADE field type and select **OK**. The Automated Data Edit Editor dialog displays with the ADE name in the **Fields** list.

5. Define field-specific properties for the new field from the available options on the tabs on this dialog.

6. On the General tab, select the arrow next to the **Edit Type** field (upper left-hand corner of the tab) to display the Automated Data Edit Properties dialog where you set specific ADE properties. Any fields you create using this ADE type will inherit these properties.

7. If you want this field to be applied to the forms in the current job and be available for other FormWare jobs, select the **Apply to Derived Fields** button. The Apply Automated Data Edit Properties dialog displays.

8. Select the **Confirm before change** check box to be prompted for confirmation at the form level before changes are made to existing fields assigned to this ADE type.

9. Select the **Apply to current forms** check box to apply all changes to fields of this ADE type in the current job.

10. Select **OK** and then **Yes** to save all form changes. If you selected the **Confirm before change** check box, you will receive a confirmation dialog for each form to which changes apply.

11. Select **Yes** on the dialog for each form that you want to be updated.

12. Select **OK** once the changes have been applied.

13. Select **Save All** on the Designer window toolbar to save your changes.

Copying ADE Properties

ADE field properties are stored as single-packed field text properties. To copy an ADE from one field to another, follow these steps:

1. Double-click on the ADE field you want to copy to display the Field Properties, General tab.

2. Select the **Text Properties** button. The Field Text Properties dialog displays.

3. Highlight the line in the file starting with ADE=
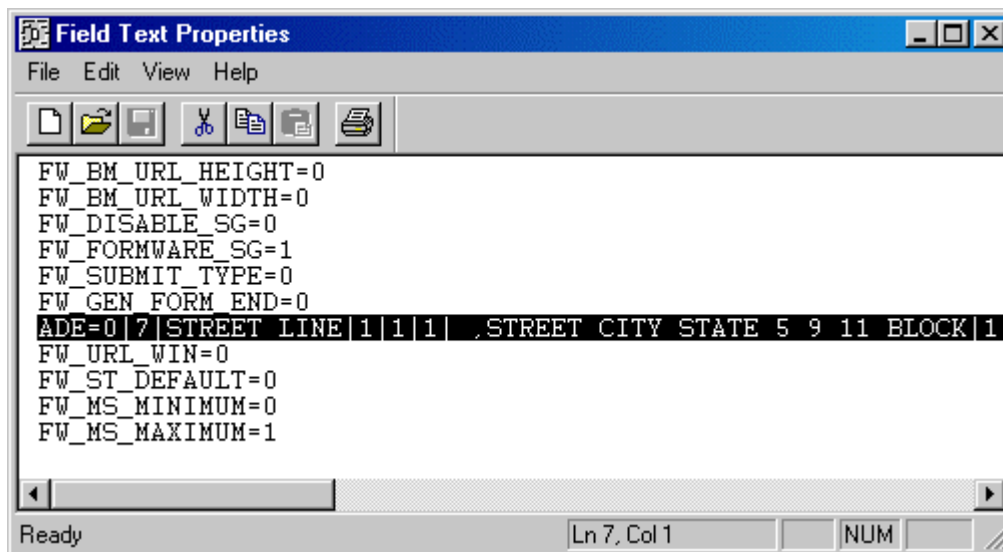
4. Select the **Copy** button.

5. Close the Field Text Properties dialog and return to Designer.

6. Select the field on the data form to which you want to copy the text properties.

7. Select the **Text Properties** button on the Field Properties, General tab for this field.

8. If there is already an **ADE** = line, remove it.

9. Paste the copied **ADE** = line into the file.

10. Save and close the file.

Note: Because the ADE specifications are text, you can also build your own text file "toolbox" of ADEs, and copy/paste from there. Including a short description before each ADE would be helpful to explain its use on the form.

Defining Address ADE Properties

If you selected **ADDRESS** on the Automated Data Edits dialog (refer to the Adding an ADE Field to a Form section for instructions), define properties for this ADE type as follows:

1. Select the arrow next to the **Edit Type** field on the Field Properties, General tab for the Address field. The Automated Data Edit Properties, Format tab displays.

Note: During Recognition, these formats are enforced only after any extra data allowed by the Recognition Formats has been removed.

---

2. On the Format tab, select from the following format types:

      o **Street Line** - Allows only a street address line in this field (e.g., 123 Main St., PO Box 45, 23 West 4th Ave, Apt 12).

      o **City** - Allows only a city in this field (e.g., Park City, San Diego).

      o **State -** Allows only a state in this field (e.g., California, CA).

      o **Zip (5, 9 or less, 11 or less)** - Allows only a zip code in this field (e.g., 12345, 12345-5678, 12345 5678).

      o **City, State, Zip (5, 9 or less, 11 or less)** - Allows a city, state, and zip code combination in this field (e.g., Waltham, MA 02451).

o **Block (Zip 5, 9 or less, 11 or less)** - Allows an entire address in this field (e.g., 123 Main St., Waltham MA 02451).

Note: You can split the City, State, and Zip and Full Address Block formats across multiple lines on the form.

---

3. Select the Validation tab for the Address ADE type.

4. Select the **Use Lookup Table** check box to type or **Browse** for a validation parameter different from the default. Lookup tables contain a list of valid input possibilities. Validation compares the data input in the field to the list of possible values. If a match is not found, the field is rejected. If you do not type the path to the lookup table file, FormWare assumes the file resides in \FormWare\Rules folder.

5. In the **Table Contains** area, select the check boxes for the address components that exist in the lookup table (i.e., **Cities, States, Zips**). If you select the **Zips** check box, also type the number of Zip code digits allowed in the table. Each entry in the table must be built using the checked components, in the order listed below. For example, if you select all options, and you specify five digits for the Zip code, then validation expects the value for Waltham, MA 02451 to be WALTHAMMA02451 in the lookup table. If you do not select State, then validation expects WALTHAM02451.

6. In the **Not before** field in the **Zip Code Range** area, type the lowest number in the range to which Zip code entries will be compared during validation. Zip code entries lower than this value will be rejected.

7. In the **Not after** field, type the highest number in the range to which Zip code entries will be compared during validation. Zip code entries higher than this value will be rejected.

8. Select the Recognition tab for the Address ADE type.

9. In the **Formats** area, deselect the address formats that you do not want to use during Recognition.

10. In the **Address Block** area, select the **Retain line(s) above street** check box to recognize lines above the street line.

11. Select the **Use PerfectPost** check box to use the PerfectPost database for address validation. The third-party PerfectPost database, which can be purchased as an add-on component, contains known street, city, state, and zip code addresses within the United States.

12. In the **Normalize Spellings** area, select the **Cities** and\or **States** check boxes to uses standardized spellings of cities and states and compare them to the lookup tables you specify.

13. Select **OK** to save your settings and return to the Field Properties dialog.

Defining Amount ADE Properties

If you selected **AMOUNT** on the Automated Data Edits dialog (refer to the Adding an ADE Field to a Form section for instructions), define properties for this ADE type as follows:

1. Select the arrow next to the **Edit Type** field on the Field Properties, General tab for the Amount field. The Automated Data Edit Properties, Format tab displays.

2. On the Format tab, select from the following format types:

    o **DDD** - Allows one or more whole-number digits (e.g., "dollars" in U.S. format).

    o **CC** - Allows *exactly* two fractional digits (e.g., "cents" in U.S. format).

- o **dddCC** - Allows 0 or more whole-number digits (ddd) and *exactly* two fractional digits (CC).
- o **DDDcc** - Allows one or more whole-number digits (DDD) and 0 or *exactly* two "fractional" digits.
- o **DDDCC** - Allows one or more whole-number digits (DDD) and *exactly* two fractional digits (CC).

Note: During Recognition, these formats are enforced only after any extra data allowed by the Recognition Formats has been removed.

3. Select the Validation tab for the Amount ADE type.

4. In the **Minimum** field in the **Integer Validation** area, type the minimum value to flag the field if any integer portion is less than this amount. Leave this field blank for no minimum value validation check.

5. In the **Maximum** field, type the maximum value to flag the field if any integer portion is greater than this amount. Leave this field blank for no maximum value validation check.

6. In the **Fraction Validation** area, select the **Perform fraction validation using table** check box, to reject a field that requires cents if the cents digits are too few. Otherwise, it assumes dollars (i.e., that optional cents are empty). Select the name of the table to validate against if the fractional value is part of the Recognized data format or Output format.

7. In the **Table has** area, select whether the lookup table has **Valid** or **Invalid** fractional amounts. For amount validation leading and trailing spaces are ignored. For example, a field having *$23.00* stored as *" 2300"* would be validated as if it contained *2300*.

8. Select the Recognition tab for the Amount ADE type.

9. In the **Formats** area, deselect the amount formats that you do not want to use for Recognition.

10. Select the **Allow optional commas** check box if the field allows commas (e.g., 65,000).

11. In the **Decimals on Form** area, select whether decimal values are **Required** or **Optional**.

12. In the **Money Symbols on Form** area, select if money symbols are **Required**, **Never**, or **Optional** on form. If the symbol is other than the $ sign, type it in the field provided (e.g., £).

13. In the **Pad Missing Fractional Amounts** area, select the **Pad integers less than** check box to allow certain amounts that may have fractions. Type the pad parameters in the field provided.

14. Select **OK** to save your settings and return to the Field Properties dialog.

Defining Date ADE Properties

If you selected **DAY** on the Automated Data Edits dialog (refer to the Adding an ADE Field to a Form section for instructions), define properties for this ADE type as follows:

1. Select the arrow next to the **Edit Type** field on the Field Properties, General tab for the Date field. The Automated Data Edit Properties, Format tab displays.

2. On the Format tab, select from the following format types:
    - o **DD, MM, CC,** or **YY** - Allows 2-digit day, month, century (20), or year (01)
    - o **DDMM**, **MMDD**, **MMYY**, **ccYY**, or **CCYY** - Allows day and month or month and day, month and 2-digit year, 2-digit year, or 4-digit year.

- o **DDMMYY**, **MMDDYY**, **MMCCYY**, or **MmccYY** - Allows day, month, and 2-digit year; month, day, and 2-digit year; month and 4-digit year; or month and 2-digit year.
- o **DDMMccYY**, **DDMMCCYY**, **MMDDccYY**, or **MMDDCCYY** - Allows day, month, and 2 or 4-digit year; day, month, and 4-digit year, month, day, and 2 or 4-digit year; or month, day, and 4-digit year.

Note: During Recognition, these formats are enforced only after any extra data allowed by the Recognition Formats has been removed.

3. Select the Validation tab to specify validation for a time frame relative to today, a specified day, or according to specific user-defined ranges (day, month, year, and century).

4. In the **Max Days Before** field in the **Relative to Today** area, type the number of days prior to the current date for which corresponding date input can be accepted. For example, if on 12-1-02 you type a 5, a date field input of 11-29-02 would be accepted, but 11-21-02 would not. This number is inclusive of the current date. For example, if you type a 3, a date entry for today and one or two days prior to today would be accepted; a date entry for three days prior to today would be rejected. If you leave this field blank, no date input validation occurs.

5. In the **Max Days After** field, type the number of days after the current date for which corresponding date input can be accepted. For example, if on 12-1-02 your input in this field is 5, a date field input of 12-04-02 would be accepted, but 12-17-02 would not. This number is inclusive of the current date. If you leave this field blank, no date input validation occurs.

6. In the **Not Before** field in the **Relative to Specific Dates** area, type the date against which entries in the field are compared. Date input before the date you specify is rejected. The date input in this field must be in MMDDCCYY format (i.e., the European date format is not acceptable for this option). Leave this field blank to prevent date input validation.

7. In the **Not After** field, type the date against which date entries in this field are compared. If the date input is after this date, the field is rejected. The date input in this field must be in MMDDCCYY format. Leave this field blank to prevent date input validation

8. In the **Specific Ranges** area, type the values that establish validation time frames based on minimum and maximum ranges for day, month, year, and century. Values outside of the range will be rejected.

Note: FormWare ensures that you can select only "input" formats that will work with the designated "output" format. Because the set-ups for **input** (recognition) and **output** (format) on a particular field must work together, the set of available input formats on the Recognition tab will depend on the output selected on the Format tab.

9. Select the Recognition tab for the Date ADE type.

10. In the **Formats** area, deselect the formats that you do not want to use during Recognition.

11. In the **Types of Months** area, select the type of values to accept for months during Recognition. You can check one or both of these values. Options are:
    - o **Alpha** – Enable Recognition to accept alpha input values for the month portion of the date.
    - o **Numeric** – Enable Recognition to accept numeric input values for the month portion of the date.

12. In the **Autofill Missing Century** area, set parameters that determine how field autofill is handled. FormWare automatically populates blank fields with the data you specify. There is no maximum number of characters for these fields. Select from the following options:

o **Do Not Autofill** - The missing 2-digit century should remain blank.

o **By Value** - Defaults to 20 (the current century). Change to "19" if Recognition applies to the previous century.

o **Imply from Year Ranges** - FormWare *Y2Kparms* supports the auto generation of century, based on the year. The Y2Kparms string consists of 4 values: low1900, high1900, low2000, high2000. For example, *Y2Kparms=90,99,00,50* indicates that if the year is between 90 and 99, the century value is "19." If the year is between 00 and 50, the century value is "20." And, if the year is between 51 and 89, the field is rejected because no year can be implied.

13. Select the **Pad Single Digit Months and Day** check box to convert single-digit date input to double digit by inserting a zero where needed.

14. Select **OK** to save your settings and return to the Field Properties dialog.

Defining Name ADE Properties

If you selected **NAME** on the Automated Data Edits dialog (refer to the Adding an ADE Field to a Form section for instructions), define properties for this ADE type as follows:

1. Select the arrow next to the **Edit Type** field on the Field Properties, General tab for the Name field. The Automated Data Edit Properties, Format tab displays.

2. On the Format tab, select from the available format types. Format parameters for Name ADE include combinations of **First**, **Last**, **Middle** names, with **Title** or **Profession** or just **Title** or **Profession**.

Note: During Recognition, these formats are enforced only after any extra data allowed by the Recognition Formats has been removed.

3. Select the Validation tab for the Name ADE type field.

4. Select the validation parameters that indicate whether to use table lookups for:

o **Use First/Middle Lookup** -use the FNames.tbl or other lookup table for first/middle name.

o **Use Last Lookup** -use the LNames.tbl or other lookup table for last name.

o **Use Title Lookup** -use the Titles.tbl or other lookup table for title.

o **Use Profession Lookup** -use the Profesns.tbl or other lookup table for profession.

Use the **Browse** button to locate the appropriate lookup table (in the FormWare\Rules folder), which contains a list of valid inputs. Validation compares the data in the field to the list of possible values and rejects the field if no match is found. You can also indicate the maximum number of characters in the adjacent **Length** field for each parameter in the lookup table. If a name to be validated is longer, only this number of characters will be compared to values in the table.

5. Select the Recognition tab for the Name ADE type.

6. In the **Formats** list, deselect the formats for which you do not want to perform Recognition.

7. In the **Truncate** area, select whether you want portions of the name format (**First**, **Middle**, **Last**) to be truncated to a maximum length on output. Select the appropriate check box and specify the maximum desired length. If you specify a length different from a maximum length specified on the Names/Validation tab for lookup tables, that length should be a longer maximum.

8. In the **Ambiguous names** area, select **Yes** or **No** to determine if the order of names can be reversed on the form. The lookup tables specified on the Names/Validation tab can be used to determine the order and to enable the desired output name order. If, for example, the output format is "FIRST LAST," one form might show *Lindsay Taylor* and another *Taylor Lindsay*. If you select **Yes**, *Taylor Lindsay* would be output correctly as *Lindsay Taylor*.

9. Select **OK** to save your settings and return to the Field Properties dialog.

Defining Phone ADE Properties

If you selected **PHONE10** or **PHONE7** on the Automated Data Edits dialog (refer to the Adding an ADE Field to a Form section for instructions), define properties for this ADE type as follows:

1. Select the arrow next to the **Edit Type** field on the Field Properties, General tab for the Phone ADE field. The Automated Data Edit Properties, Format tab displays.

2. On the Format tab, select from the formats listed. The format list shows phone-format separators for clarity, but they are NOT required for input, nor will they be output. The international standard is **country code > city code > number.** For example:

    o   Germany: 49 761 70 324 100

    o   UK: 44 0 190 884 3649

    o   Australia: 61 3 9811 9933

    Internationally, the names as well as the lengths of each of the parts can be changed. Therefore, any phone number format having 1 to 5 parts can be accommodated. In the following US formats, *C* is country, *A* is area code, *E* is exchange (first 3 digits of 7-digit phone number), *N* is number, and *x* is extension.):

    o   **CCC** - Required country code (3 digits).

    o   **AAA** - Required area code (3 digits).

    o   **X** - Standalone extension (any number of digits).

    o   **EEE-NNNN** - Three exchange digits and four-digit telephone number required.

    o   **(aaa) EEE-NNNN** - Optional area code, required exchange digits and four-digit telephone number.

    o   **(AAA) EEE-NNNN** - Required area code, exchange digits and four-digit telephone number.

    o   **(aaa) EEE-NNNN x** - Exchange digits and four-digit telephone number required; optional area code and exchange.

    o   **(AAA) EEE-NNNN x** - Required area code, exchange digits and four-digit telephone number; optional extension.

    o   **ccc (AAA) EEE-NNNN** - Optional country code; required area code, exchange digits, and four-digit telephone number.

    o   **CCC (AAA) EEE-NNNN** - Required country code, exchange digits and four-digit telephone number.

Note: In the international context the number of phone-format characters (C, c, A, a, E, or N) will likely differ from the USA style. During Recognition, these formats are enforced only after any extra data allowed by the Recognition Formats has been removed.

---

3. Select the Validation tab for the Phone ADE type.

4. Select the **Use Lookup Table** check box and type or **Browse** to the lookup table (in the \FormWare\Rules folder) for phone number validation. Leave this field blank or no phone validation.

5. In the **Table Contains** area, select the check boxes that indicate the phone number components contained in the lookup table: **Country Codes, Areas Codes**, and\or **Exchange Codes**.

6. Select the Recognition tab for the Phone ADE type.

7. Deselect any phone formats that you do not want to use during Recognition.

8. Select **OK** to save your settings and return to the Field Properties dialog.

Defining Social Security Number (SSN) ADE Properties

If you selected **SSN** on the Automated Data Edits dialog (refer to the Adding an ADE Field to a Form section for instructions), define properties for this ADE type as follows:

1. Select the arrow next to the **Edit Type** field on the Field Properties, General tab for the Phone ADE field. The Automated Data Edit Properties, Format tab displays.

2. On the Format tab, the one supported Social Security Number format displays. SSN is a single format and thus entails no option for input (i.e., Recognition). Select this option to continue.

3. Select the Validation tab for the SSN ADE type.

4. Select the **Use Lookup Table** check box to use a lookup table (in the \FormWare\Rules folder) to validate Social Security Numbers.

5. Type or browse to add the name of the SSN lookup table.

6. Select **OK** to save your settings and return to the Field Properties dialog.

Defining Time ADE Properties

If you selected **HOURS, MINUTES,** or **SECONDS** on the Automated Data Edits dialog (refer to the Adding an ADE Field to a Form section for instructions), define properties for this ADE type as follows:

1. Select the arrow next to the **Edit Type** field on the Field Properties, General tab for the Time ADE field. The Automated Data Edit Properties, Format tab displays.

2. Select Time for the ADE field type and open the Automated Data Edit Properties dialog.

3. On the Format tab, select from the format types. In the following formats, *S* is seconds, *M* is Minutes, and *H* is Hour.
    o   **SS**, **MM**, or **HH** - Required 2-digit seconds, minutes, or hour.
    o   **MMhh** or **hhMM** - Required minutes, optional hour.
    o   **mmHH** or **HHmm** - Optional minutes, required hour.
    o   **MMHH** or **HHMM** - Required minutes and hour.
    o   **hhMMSS** - Optional hour, required minutes and seconds.

    Continue to read the formats as indicated by the letters in the order you want and whether required (uppercase) or optional (lowercase).

> **Note**: During Recognition, these formats are enforced only after any extra data allowed by the Recognition Formats has been removed.

4. Select the Validation tab for the Time ADE type.

5. In the **Hours** area, type the **Minimum** or **Maximum** acceptable hours for this field. For example, allow entries of at least 24 hours or none over 72 hours.

6. In the **Minutes** area, type the **Minimum** or **Maximum** acceptable minutes. For example, allow entries of at least 60 minutes or none over 180 minutes.

7. In the **Seconds** area, type the **Minimum** or **Maximum** acceptable seconds. For example, allow entries of at least 60 minutes or none over 180 minutes.

8. Select the Recognition tab for the Time ADE type.

9. Deselect the time **Formats** that you do not want to process during Recognition.

10. Select the **Separator Required** check box to indicate that a colon is required between hours and minutes.

11. In the **Hours Format** area, select whether to use a 12-hour or 24-hour time format.

12. In the **12 Hour AM/PM Indicator** area, select the **Indicator Required** check box if you are using a 12-hour time format. Then, select an AM and\or PM indicator to display with the time if desired, e.g., 10:15 A (or AM or A.M.).

13. Select the **Pad Single Digit Hours and Minutes** check box to pad minute and hour designations with zeroes.

14. Select **OK** to save your settings and return to the Field Properties dialog.

Using the Automated Data Edit Editor

1. To select from the complete list of ADE field types and set properties, select the **Automated Data Edit Editor** button on the Field Types toolbar. The Automated Data Edit Editor dialog displays all pre-defined and user-defined fields on your system. When you select a data field from the field list, its field properties display in the right portion of the editor. Only the properties that can be modified for user-defined fields within the editor are enabled.
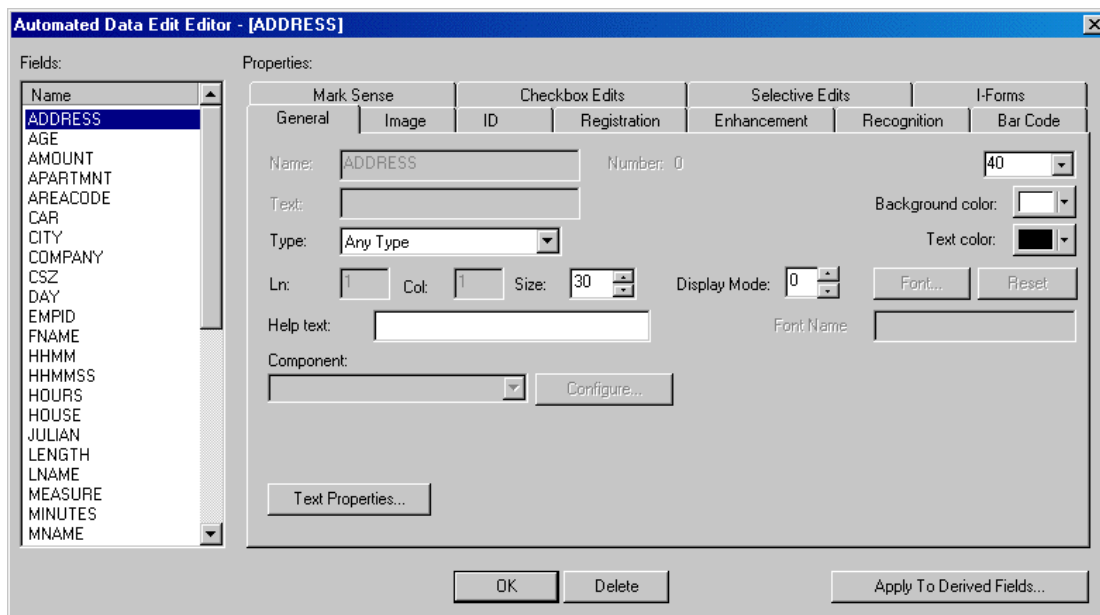
2. Follow the steps in the other field property sections to define ADE properties. Some options are disabled based upon the field type you select from the list at the left side of the dialog.

Understanding the Processing Order for ADEs

FormWare Designer supports form creation that includes ADEs and processing within the Recognition and Completion modules. ADEs process in the following order:

- Recognition Processing
  - o 1. ADE field is recognized
  - o 2. Raw results are returned
  - o 3. Free form parameters applied
  - o 4. Rules applied
  - o 5. ADEs applied
  - o 6. VBA applied
- Completion Processing
  - o 1. Data keyed in field
  - o 2. Data released
  - o 3. Checkbox edits run
  - o 4. ADEs run
  - o 5. FPL code run
  - o 6. VBA code run

Using ADE Lookup Tables

A number of lookup and translation tables exist that contain a list of possible input values for various ADEs. During validation, data in a given field are compared to the list of possible input values in the lookup table. Lookups occur for address, amount, phone, SSN, and name information. In most cases, if a match is not made, the field is rejected. In some cases, the table contains a list of prohibited entries. In those cases, if a match *is* made, the field is rejected.

Translation tables contain conversion information for data within field input that must be converted to another format before being output. Translations occur primarily for address information.

Lookup and translation tables are text files that reside in the \FormWare\Rules folder. You can modify these tables to suit your specific needs. Use a standard text editor to view and edit the tables.

Caution: Some editors, such as MS-Word and WordPad, add hidden extra characters to the file. Prevent this by ensuring the file is saved in "plain text" format (.txt).

Contents of lookup tables should always be presorted. When you specify a lookup or translation table in any of the ADE dialogs, you need only enter the filename; the default location (\FormWare\Rules) is assumed. However, if you have moved the files, you must also include the full path for the new location. File size limitations for lookup tables are based upon the memory capacity of the PC running Completion, Recognition, and any other clients that can run VBA to invoke ADE processing.

The following lookup and translation tables are included with the FormWare software for ADEs:

- **CitiesUS.tbl -** Lookup table for address validation that contains the names of most cities and towns in the U.S.

- **StatesUS.tbl** - Lookup and translation table for address validation that includes standard two-letter abbreviations for each state in the U.S. For translation, it also contains the state names spelled out and the correct two-character code for the translated version.

- **StateZip.tbl** - Lookup table for address validation that contains the abbreviations for each state and the possible beginning two digits for zip codes in each state within the U.S.

- **AmtFract.tbl** - Lookup table for amount validation that contains all fractional amounts that can be present as amount field entries.

- **Titles.tbl** - Lookup table to validate titles for name fields. This table lists possible titles alphabetically. Titles should appear in uppercase.

- **Profesns.tbl** - Lookup table to validate professions for name fields. This table lists possible professions alphabetically. Professions should appear in uppercase.

- **Fnames.tbl** - Lookup table to validate first and middle name for name fields. This list is alphabetical. Names should appear in uppercase.

- **Lnames.tbl** - Lookup table to validate last name for person's name fields. This table lists possible last names alphabetically. Names should appear in uppercase. This table contains a large, but not complete, set of last names. You can regenerate this table using your own resources (e.g. mainframe dump) to get a more accurate table for your applications.

Creating a City Lookup Table

The goal of this lookup table is to validate entries against a set of existing city names. Create a lookup table file that includes the target city names. Name the file with a .tbl extension and store it in the \FormWare\Rules folder. Example entry formats in the lookup table are:

```
WALTHAM

LOS_ANGELES
```

Note: Entries must be uppercase and internal spaces must be replaced with the underscore character. Substitutions are also allowed.

For example, if the official city name is North Bluff, and you also want N Bluff to be accepted after translating to the official name, then the two entries would be:

```
NORTH_BLUFF
N_BLUFF NORTH BLUFF
```

Creating a Phone Number Lookup Table

The goal of this custom phone lookup table is to validate phone number entries against a set of existing phone numbers. Name the file with a .tbl extension and store it in \FormWare\Rules folder. Using a sample number 0110-212-932-3456 (where *0110* is a country code, *212* is an area code, and *932* is an exchange), entries in the table would be:

```
0110 – Country code only
212 – Area code only
932 – Exchange only
0110212 – Country code and area code
0110932 – Country code and exchange
212932 – Area code and exchange
0110212932 – All (Country, Area, and Exchange codes)
```

Any specified output separators are excluded from the lookup. Also, definitions of each entry are included for clarity only—they should NOT be in the actual .tbl file.

Creating a SSN Lookup Table

The goal of this custom Social Security Number (SSN) lookup table is to validate SSN entries against a set of existing Social Security Numbers. Create a file with these numbers and name it with a .tbl extension. An example entry format in the lookup table is:

```
242991212
```

Any specified output separators are excluded from the lookup.

Understanding Image Pre-Processing

The Image Pre-Processing component (IPP) cleans up your images to make them "recognition ready." IPP is a necessary, but often overlooked part of any forms processing system. It can eliminate many of the problems associated with forms processing applications that use recognition technologies on images. Pre-processing your images can improve image readability and, ultimately, the data recognition accuracy rate of your images.

The most important features of the IPP component are image deskew and registration. Image deskew examines lines or text on the image, determines if the image is skewed, and if so, straightens the image. Deskew can improve accuracy of the recognition engine to a limited degree, but registering the image affords a greater degree of recognition accuracy.

Image registration provides a consistent upper and left margin for each image. By forcing all images to have a uniform upper and left margin, the recognition engine can detect image recognition zones with more precision. When you have a well-designed paper form, image deskew and registration are the only features you need for precise recognition. However, if your paper forms are not designed for use with recognition technologies, you may need to apply Image Enhancement options.

In addition to deskew and registration, IPP can remove lines that contain gaps and reconstruct characters that intersect lines to maintain the text for recognition. To prevent unreadable images from proceeding to the recognition process, IPP can detect upside-down images and rotate them

to the correct orientation. It can also rotate images that are scanned at different orientations to avoid an edge perforation and ensure that the image displays correctly on the screen.

You can also define IPP settings for color images. See the Understanding Color Image Pre-Processing section for more information.

The IPP component is fully integrated with JobFlow. Within a JobFlow, batches are routed to pre-processing, and then to the next task, such as Form Identification, Image Enhancement, or Recognition. As part of the JobFlow system, IPP can function as an unattended task, eliminating the need for an operator.

Caution: IPP has many options available to improve the readability of your images. Be aware, however, that any time you select a feature, the time required to process the image is increased. If you process a small number of images, this may not matter, but if you process a high volume, a few extra seconds can make a big difference. Also, performing this time-intensive step may not improve the recognition accuracy rate.

By performing a cost-benefit analysis on your image processing procedures, you can determine the advantages and the disadvantages of pre-processing your images in terms of the time required. A rule of thumb is to use only those features that produce the cleanest image possible in the shortest amount of time.

You can define the following IPP settings:

- Job-Level IPP Properties
- Form-Level IPP Properties
- Field-Level IPP Properties

The following table lists which IPP processes occur at which level: Job, Form, or Field.

Level	Tab - Task	Function
Job	IPP – Properties	Set Job level IPP properties, including color.
Job	Test – IPP	Test Job level IPP against a batch of images.
Form	IPP – Properties	Set Form level IPP properties to apply during Enhancement.
Form	Image – IPP	Runs Job level IPP on the setup image.
Form	Test– IPP	Test Job level IPP against a batch of images.
Form	Test – Enhancement	Test Form level IPP against a batch of images.
Field	IPP – Various Removal Options	Set Field level removal options to apply during Enhancement.
Field	IPP – Register/Deskew	Set Field level IPP to apply during Enhancement.
Field	IPP – Test	Test Field level Enhancement on setup image.
Field	Test – Enhancement	Test Form level IPP against a batch of images for one field.
Field	Test – IPP	Test Job level IPP against a batch of images.

Understanding Color Image Pre-Processing

Using Color Image Pre-Processing (IPP) you can display color JPG images in Designer, Completion, and in the Image Viewer. During IPP, FormWare can automatically detect the background color of the image by analyzing the spectrum of colors in the image and deriving the average color. You can run IPP against a color setup image.

For more overview information on color images, see the Working with Color Images section.

The Color IPP process inserts a parameter into the BDF with the background color value (i.e., BackVal=RGB Radius) for each color image it processes. To enable this function, select the **Enable Virtual Form ID tags** check box on the Image Pre-Processing Properties, Color tab. The Auto-Training process reads this parameter from the BDF and uses it for training pre-ID color values for each new sub-class.

Color IPP performs the following tasks:

- Automatically detects and removes colored backgrounds.
- Converts color images to black and white prior to OCR.
- Performs deskew and border-cropping of color JPG images.

When using color IPP, you should be familiar with the following terms:

- RGB - Standard color space used by monitors and other color devices
- HSV - Hue, Saturation, and Volume color space

Color IPP support exists within several FormWare components. Most of these components do not actually process the image, but because the images were skipped, record processing for these images was also skipped. Color image support capability exists for the following FormWare components:

- DataView
- FNXClnt
- IMRClnt
- INPClnt
- ODBClnt
- OPTClnt
- OTGClnt
- RBEClnt
- VBAClnt
- XMLClnt

BDF File Modifications for Color Images

Color IPP only supports processing of JPG color images. The FW_IMT=COLOR parameter in the BDF indicates whether an image is a color image. FormWare Scan writes out the FW_IMT=COLOR parameter on the line in the BDF just preceding any color image. Bi-tonal images have no FW_IMT tags. If the BDF contains levels tags and if all of the images within a given level are color, then the Scan module writes only a single tag at the beginning of each level, rather than tagging every image.

Understanding How Color Processing Works

Note: Form-level IPP does not support color processing. All color processing is performed at the job level (using IPPClnt.ocx), prior to the normal binary image processing currently available in the IPP client.

The following sequence illustrates how color IPP occurs:

1. **Color image is scanned and forwarded to IPP.**

   FormWare detects a string that represents the background color of the image. When a color image is identified, IPP performs the following functions:

   o Performs deskew and cropping functions on the color image (optional). The original input color image gets overwritten with the corrected one.

   o Analyzes the color image to determine its color spectrum and profiles the image to determine the color distribution of the image background.

   o Removes the background color from the entire color image (optional). Removing the background color neither modifies nor overwrites the contents of the color image. It is used only as an input to the bi-tonal image conversion.

   o Converts the color image to an output bi-tonal image.

2. **Job-level binary processing is performed on the new bi-tonal image.**

   Job-level binary processing is performed on the new bi-tonal image and IPP property settings (set at the job level) are applied. This is traditionally the first IPP step for all non-color images.

3. **BDF is modified with new data about the color image.**

   The contents of the BDF data at this stage can include information on all color images, all bi-tonal images, or some combination of both. When the image is scanned using the FormWare Scan module, the BDF is generated for use by IPP. The IPP process modifies the BDF contents following color processing. The following processes create and modify the BDF file:

   o The scanning process generates the initial BDF file and tags each JPG color image with the FW_IMT=COLOR parameter.

   o The IPP process determines the background value (BACKVAL= parameter) and inserts this tag in the BDF for use by the Form ID and Auto-Training WorkFlow steps.

4. **Color image (JPG) is converted to bi-tonal image (TIFF).**

   Prior to recognition (OCR), the IPP process converts all JPG color images to in-memory, bi-tonal images. The IPP client processes the input color JPG and outputs a new bi-tonal TIF. This conversion includes the following steps:

   o The original input color image (JPG) is preserved for display in Completion.

   o If deskew or border cropping is enabled for the color image, IPP overwrites the original color image (JPG) with the corrected image (JPG).

   o When IPP successfully creates a bi-tonal (black and white) TIF image from the color image, the Designer module displays the two images in a split-window viewer, showing the image pair (bi-tonal and color) side by side.

   o If the IPP client fails to create the bi-tonal TIF, Designer displays the color image in a single-image viewer without error.

   o The bi-tonal image preserves deskew and cropping changes for use during image enhancement and recognition.

5. **Apply color threshold settings to optimize image enhancement and recognition.**

   Use the settings in the Thresholding section on the Image Pre-Processing Properties, Color tab to improve the enhancement and recognition processes. Refer to the Setting Color Properties section for more information.

6. **Write-out bi-tonal image to disk for use in later processes.**

   IPP writes the bi-tonal image to disk for use in Enhancement, Recognition, Completion, and Auto-Training. The new bi-tonal image is stored in either the same directory as the input color image or a parallel directory, depending on the input image extension. For example:

   - **Input image (JPG)**   Output bi-tonal image is written to the same directory with the same base image name and a .tif extension. For example:

     Input image name: \FormWare\Images\Image1.jpg

     Output bi-tonal image name: \FormWare\Images\Image1.tif

7. **Image is processed through preliminary Form ID**

   Color IPP outputs the form to the Form ID component. Form ID reads the color values (*BackVal*) received from color IPP and performs preliminary identification (color pre-ID) on the form.

   For additional information on training color images within Dynamic Form ID, see the Editing the TRN File for Color Image Processing section.

Note: To use the same image for color capture as for OCR/ICR would require a 300-dpi color image much larger than the 100-dpi images and a slow scanner speed.

---

8. **Image is processed through Virtual Classification (Form ID)**

   The enhanced Form ID component, Virtual Classification, uses information output from the Color IPP process to process the color image. Through Color IPP, two new processes are at work in Form ID.

   - **Pre-Form ID** - Auto-Training trains an image with the background color settings that IPP wrote to the BDF. Pre-Form ID reads the color pre-ID settings and attempts to classify an untagged image (i.e., one without a Form ID number and/or variant ID number) within the input BDF.

   - **Color pre-ID** - Form ID finds the background color tag in the input BDF and checks it against existing color values in the sub-class database. All matches that fall within the threshold limits defined on the Image Pre-Processing Properties, Color tab are deemed successful.

   Form ID uses these sub-classes in recognition (i.e., fuzzy search ID). Form ID processing continues only if one or more sub-classes exceed the threshold. If Form ID processing terminates, the form is rejected with an unknown form 99 ID.

9. **The image is processed through Recognition (OCR).**

   OCR uses the binary image, because the lower-resolution color images are approximately the same size as the 300-DPI binary images. With a 100- or 150-DPI color image, recognition results degrade significantly. In fact, even a color image at the threshold of a binary image and scaled to 300 DPI is ill-suited for quality OCR/ICR. Character images would be degraded by an increase in touching characters, filled-in holes, and noise.

Auto-Training for Color IPP

The Auto-Training process reads in the background color tag information from the BackVal= parameter in the BDF to train pre-ID color values for each new sub-class. For each sub-class, Auto-Training automatically stores a color tag in the VFID database. The process works as follows:

1. Auto-Training assesses each image in the input BDF to decide whether this image should be the base image for a new sub-class or variant.

2. Auto-Training exempts images that are pre-tagged with either a normal Form ID number or a Variant ID number.

3. If, from the Completion data, Auto-Training decides this image represents a new sub-class or variant, the BackVal parameter in the BDF is checked for the background color value.

4. Existing color values are stored as part of the new sub-class or form variant pre-ID parameters.

5. The color string is written into the VFID database for that sub-class entry for use during Form ID.

6. Later in the WorkFlow, Form ID uses the new sub-class entry to identify the form.

Converting Output Color Images to a Specific Scaling (DPI)

You can convert color images to a specified output DPI based on the following settings:

1. Check the **Convert Image DPI (FAX)** option on the Image Pre-Processing Options dialog (accessed through the Job Properties, IPP page within Designer).

2. Set the INI file parameter **FAXCVTDPI=<OutputDPI>**, where **OutputDPI** is the desired Dots Per Inch (DPI) output value.

When you set these values prior to converting a bi-tonal TIF image, the color output image DPI resolution setting will correspond to the bi-tonal output image resolution setting.

This now allows you to process 200 DPI color images and correctly convert and display those images as 300 DPI bi-tonal images and 300 DPI color images.

Setting Job-Level IPP Properties

Job-level IPP performs a general cleanup of images. Setting IPP processes at this level will apply to all images in your job, even if your job contains mixed-form documents. Before you apply IPP at the job-level, evaluate your job to see if any setting would be better applied at the form-level.

Establishing Job-Level IPP Properties

To set IPP properties for the job:

1. Open the job in Designer.

2. Select the **Properties** button. The Job Properties tab displays.

3. Select the IPP tab.

4. Select the **Properties** button on the IPP tab. The Image Pre-Processing Options dialog displays.
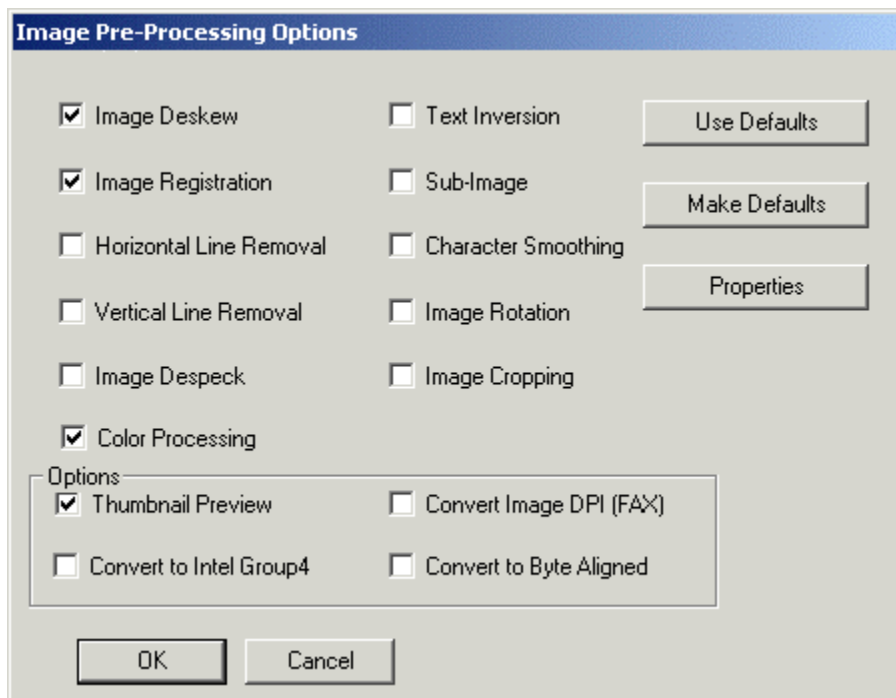
5. Select from the following check box options:
    o **Image Deskew** - Applies the settings from the Image Deskew tab, which straightens the image on the page.
    o **Image Registration** - Applies the settings on the Horizontal Registration, Vertical Registration, and Registration Control tabs.
    o **Horizontal Line Removal** - Applies the settings on the Horizontal Line Removal tab.
    o **Vertical Line Removal** - Applies the settings on the Vertical Line Removal tab.
    o **Image Despeck** - Applies the settings on the Speck Removal, Dot Shading, and Dot Verification tabs.
    o **Color Processing** - Applies the settings on the Color tab.
    o **Text Inversion** - Applies the settings on the Inverse Text tab.
    o **Sub-Image** - Applies the settings on the Sub-Image tab.
    o **Character Smoothing** - Applies the settings on the Smoothing tab.
    o **Image Rotation** - Applies the settings on the Rotation tab.
    o **Image Cropping** - Applies the settings on the Black Crop, White Crop, Border Crop, Miscellaneous Crop, and Manual Crop tabs.
    o **Thumbnail Preview** - Displays a thumbnail of the image during image processing within the Designer module. You must set your PC display to 256 colors to use this option. You cannot see thumbnails in 16-bit or 32-bit color modes. The thumbnail also does not display during JobFlow processing. Do not check this option if you use the **Convert to Intel Group4** option as it disables thumbnail generation.
    o **Convert to Intel Group4** - Converts image files received from the scanner in another format to Intel Group4 TIFF images. Since the conversion does not change

the file extension to TIF, use a three-digit numeric extension for your image files rather than the image type.

o **Convert Image DPI (FAX)** - Converts Fax images, which are typically Group3 images, into the Group4 TIFF images that the system processes. To specify the DPI to which the images are converted, you must specify the FAXCVTDPI = # parameter in the System.ini file, where # is the DPI of the converted image. The default is 200 DPI.

o **Convert to Byte Aligned** - If selected at the job level, the width of all output images produced by IPP will be rounded up to the nearest 8-bit boundary (MOD8) before being written to disk. Use this option to export images to a mainframe system that requires a byte-aligned ".TIFF" format. This option is most useful when used with the IPP deskew option because deskew will sometimes modify the width of an input source image that is byte aligned to an output image that is not byte aligned. You can then use this option to re-align the width of the output image. If selected at the form level, the previous sequence of operations is performed during Enhancement.

Note: Selecting a process from this dialog that requires further definition on the tab, triggers a warning message.

6. Select the **Properties** button to define the settings on the individual IPP property tabs. Some of these properties (but not all) can be set at the form level as well.

Note: The IPP setting you select on each tab will take effect only if the related box on the Image Pre-Processing Options dialog is checked.

7. Select the **Use Defaults** button to use the original system default settings.

8. Select the **Change Defaults** button to overwrite the system defaults with your changes.

9. Select **OK** to accept your settings and return to the Job Properties dialog.

10. Select the **Save All** button to save your IPP settings.

Setting Deskew Properties

Image Deskew straightens the image on the page. The IPP process can deskew the text on an image up to 10° horizontally or vertically. Skewed TIFF Group4 often compress an additional 25% when deskewed.

To deskew your images:

1. On the Image Pre-Processing Properties dialog, select the Deskew tab.
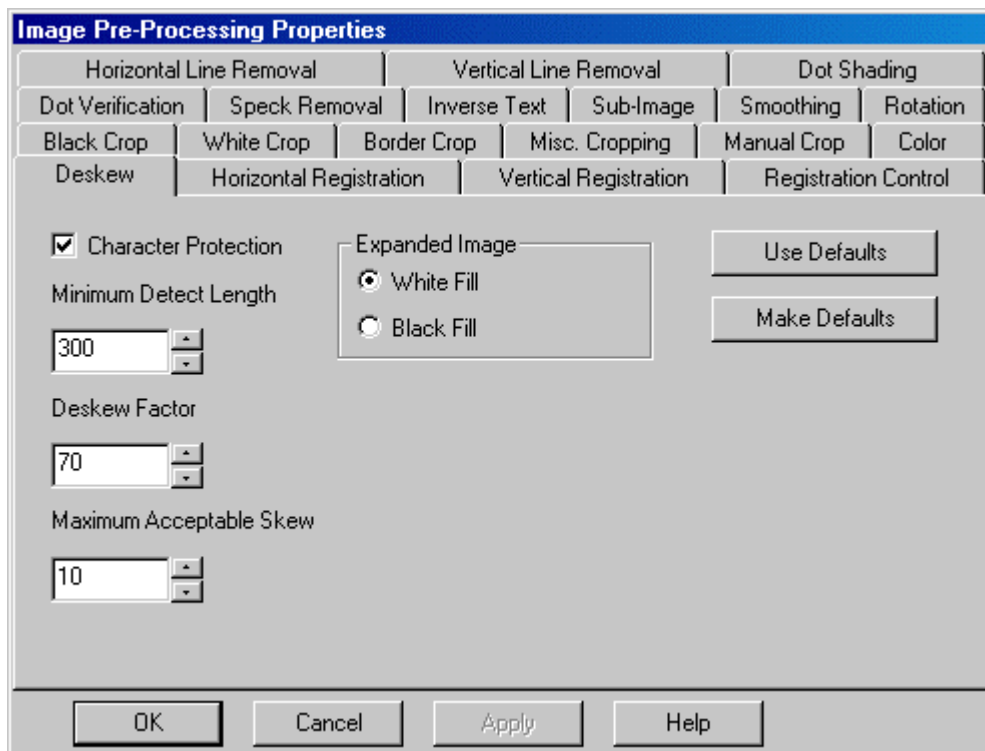
2. Select the **Character Protection** check box to reduce the normal rotational distortion of text when the image is straightened. This option is automatically disabled on images that are significantly skewed. Although character protection takes longer to process than regular deskewing, it can significantly improve the recognition accuracy rate.

3. In the **Minimum Detect Length** field, select the minimum length of a line or group of characters used to detect the skew. For example, a 3-inch line on a 300 DPI image would equal 900 pixels. The default is 300. The range is 10 – 10,000 pixels.

4. In the **Deskew Factor** field, select the deskew factor for images. This factor is dependent on the resolution of the image. For small, dense print, use a smaller numeric value such as 40 or 20. Use a larger numeric value for a large-print document such as 200 or 500. The default is 70. The range is 0 – 1,000.

5. In the **Maximum Acceptable Skew** field, select the maximum amount of skew that is acceptable for the image. Any image with a skew less than this value is not pre-processed. By reducing this value, you can decrease processing time because you are processing fewer images. The default is 150. The range is 10 – 1,000.

6. In the Expanded Image area, select the additional area fill color on the image: **White Fill** or **Black Fill**. When you deskew an image, the size of an image can increase slightly during this process. White fill is the default.

7. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

8. Select **Apply** to activate your changes.

Setting Horizontal Registration Properties

Horizontal registration provides a consistent left margin for the image. It locates the left edge of text on the image and makes this the new, left edge of the image. By using both horizontal and

vertical registration, you create a consistent left and upper margin on your image and improve the recognition accuracy rate because the recognition zones are properly aligned on each image. Most operations use only horizontal registration. However, if precise text location is important, use both processes.

When using line registration, set the value of the length of the line larger than the height of the tallest characters, but less than the actual length of the line. This helps ensure that characters like L and I are not mistaken for registration lines.

Note: Use deskew with horizontal or vertical registration unless your image is perfectly straight.

To set horizontal registration properties:

1. On the Image Pre-Processing Properties dialog, select the Horizontal Registration tab.
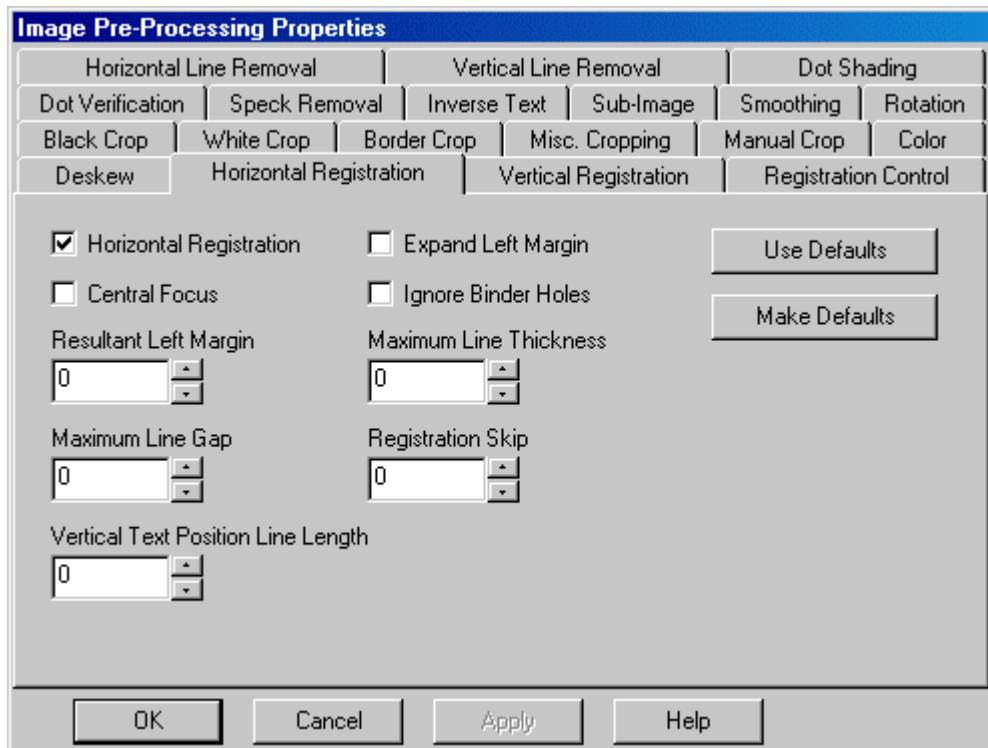
2. Select the **Horizontal Registration** check box to enable the horizontal registration fields on this tab. Horizontal registration makes the image recognition-ready by producing a consistent left margin on the image.

3. Select the **Central Focus** check box to use only the middle portion of the left border for horizontal registration. This setting is useful when the left edge contains binder holes, noise, watermarks, or logos.

4. Select the **Expand Left Margin** check box to expand the left margin during horizontal registration when the margin on the image is smaller than the value in the **Resultant Left Margin** field.

5. Select the **Ignore Binder Holes** check box to ignore binder holes and other left edge noise during horizontal registration.

6. In the **Resultant Left Margin** field, select the amount of left margin you want to keep on the image after horizontal correction occurs. A zero value moves the image text flush with the

left edge of the image. A negative value crops the image that many pixels away from the left edge. The range is -10,000 – 10,000 pixels.

7. In the **Maximum Line Gap** field, select the largest gap allowed between lines of text. An algorithm fills these spaces with black pixels and creates a virtual vertical line down the left margin of the image. Text on this page has a ¼-inch gap between lines. At 300 DPI this equals 75 pixels. The range is 0 – 10,000 pixels.

8. In the **Vertical Text Position Line Length** field, select a vertical line position length. This option bases the horizontal registration on a vertical line near the left side of the image rather than on the left edge of text. The line you use must be significantly larger than any text, particularly characters to the left of this line. This option is useful when the image contains an imprint or some other text in the left margin that is placed inconsistently. Then, instead of using the inconsistent imprinted text, the image registration occurs on this more consistent line. The range is 0 – 10,000 pixels.

9. In the **Maximum Line Thickness** field, select the maximum line thickness to use for registration. This option helps the software distinguish lines from dense text. The range is 0 – 10,000 pixels.

10. In the **Registration Skip** field, select how far from the top of the page to look for valid horizontal registration information. The range is 0 – 10,000 pixels.

11. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

12. Select **Apply** to activate your changes.

Setting Vertical Registration Properties

Vertical registration provides a consistent top margin for the image. It locates the top edge of text on the image and makes this the new top edge of the image. Most operations use only horizontal registration. However, if precise text location is important, as in sub-image extraction, use both processes.

When using line registration, be certain to set the value of the length of the line larger than the width of the characters, but less than the actual length of the line. This helps ensure that characters like T and E are not mistaken for registration lines.

To set vertical registration properties:

1. On the Image Pre-Processing Properties dialog, select the Vertical Registration tab.
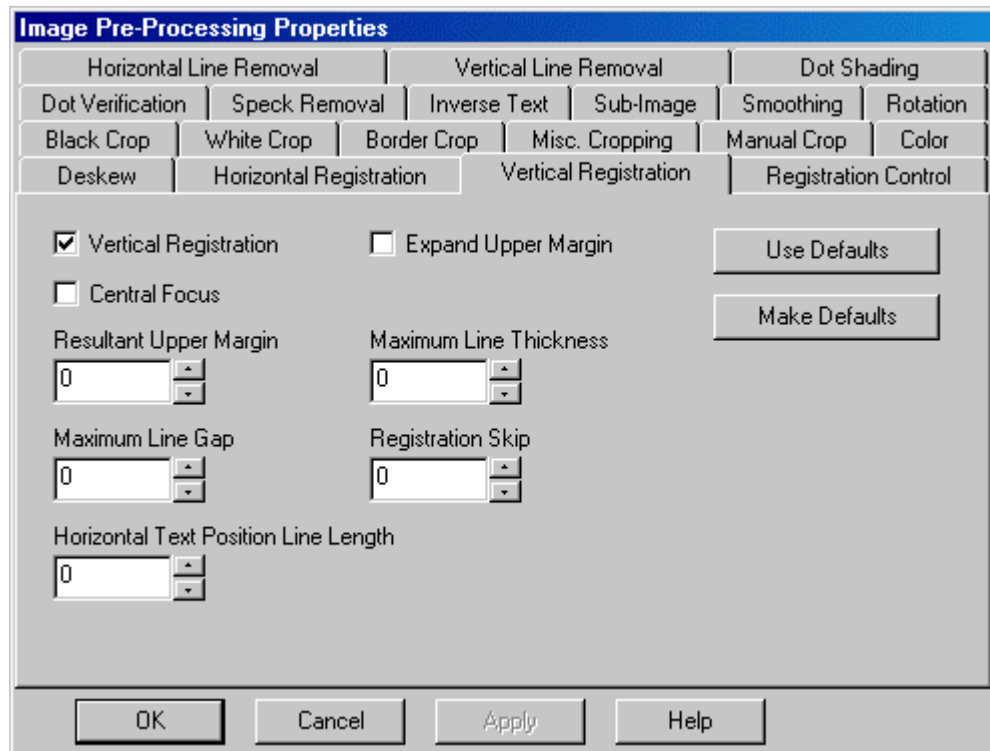
2. Select the **Vertical Registration** check box to enable the vertical registration fields on this page. This setting makes the image recognition-ready by producing a consistent top margin on the image.

3. Select the **Central Focus** check box to use only the middle portion of the top border for vertical registration. This setting is useful when the top edge contains binder holes, noise, watermarks, or logos.

4. Select the **Expand Upper Margin** check box to expand the upper margin when the current margin setting is shorter than this value.

5. In the **Resultant Upper Margin** field, select the amount of top margin to keep on the image after vertical correction occurs. A zero value moves the image text flush with the top edge of the image. A negative value crops the image text that many pixels away from the top edge. The range is -10,000 – 10,000 pixels.

6. In the **Maximum Line Gap** field, select the largest gap allowed in a horizontal line on the image. When you type a value in this field, broken segments in horizontal lines that do not exceed this value are reattached during vertical registration. A value greater than three activates a reconstruction algorithm. The range is 0 – 10,000 pixels.

7. In the **Horizontal Text Position Line Length** field, select the horizontal line length. This option bases the vertical registration on a horizontal line near the top edge of the image rather than on the top margin. The line you use must be significantly larger than any text or image data. This option is useful when the image contains an imprint or some other text in the top margin that is placed inconsistently. Then, instead of using the inconsistent imprinted text, the image registration occurs on this more consistent line. The range is 0 – 10,000 pixels.

8. In the **Maximum Line Thickness** field, select the maximum line thickness to use for registration. This option helps the software distinguish lines from dense text. The range is 0 – 10,000 pixels.

9. In the **Registration Skip** field, select how far from the edge of the page to look for valid vertical registration information. The range is 0 – 10,000 pixels.

10. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

11. Select **Apply** to activate your changes.

Setting Rotation Properties

Rotation detects whether an image is upside-down or sideways, and rotates the image accordingly. For example, due to ragged or torn edges, you scan a page in landscape orientation. When Rotation detects that the text flows up and down, it automatically turns the page so the image is in proper orientation.

Note: If all images are scanned in the same direction, it is more efficient to use the **Fixed Rotation** option to rotate them to the proper orientation.

To set rotation properties:

1. On the Image Pre-Processing Properties dialog, select the Rotation tab.
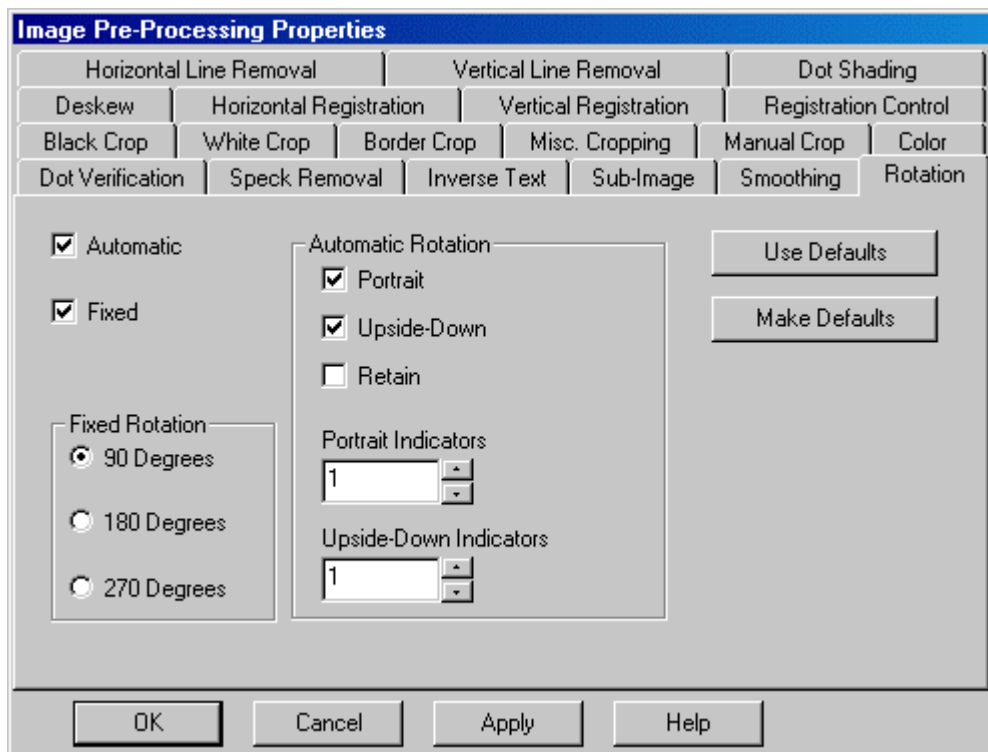
2. Select the **Automatic** check box to activate the **Automatic Rotation** settings on this tab. The software analyzes the way the text flows on the page and rotates the image to the correct position.

3. Select the **Fixed** check box to activate the **Fixed Rotation** settings on this tab.

4. In the **Fixed Rotation** area, select the degree to rotate the image. Options are **90**, **180**, or **270** degrees.

5. In the **Automatic Rotation** area, select the **Portrait** check box to detect whether the image is landscape, and rotate it to portrait.

6. Select the **Upside-Down** check box to detect whether the image is upside-down, and rotate it 180 degrees.

7. Select the **Retain** check box to return the image to its original orientation before saving it. Use this when you need to rotate the image to perform other IPP processes.

8. In the **Portrait Indicators** field, select the number of lines of text to use to verify that the image is properly oriented. For example, if you have some vertical text on the page, set this value higher to have the process look at the way several lines of text flow instead of just finding the few vertical lines on the image. If you set this value too high, it can increase the processing time. The default is 1. The range is 0 – 100 lines.

9. In the **Upside-Down Indicators** field, select the number of lines of text to use to verify that the image is not upside-down. If you set this value too high, it can increase the processing time. The default is 1. The range is 0 – 100 lines.

10. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

11. Select **Apply** to activate your changes.

Setting Black Crop Properties

The black crop feature is one of five image-cropping options within IPP. Black crop bleaches the black border that some images acquire during the scanning process. Black crop turns these black borders white, even if the border is thick and uneven. Use black crop before you use white crop.

To set black crop properties:

1. On the Image Pre-Processing Properties dialog, select the Black Crop tab.
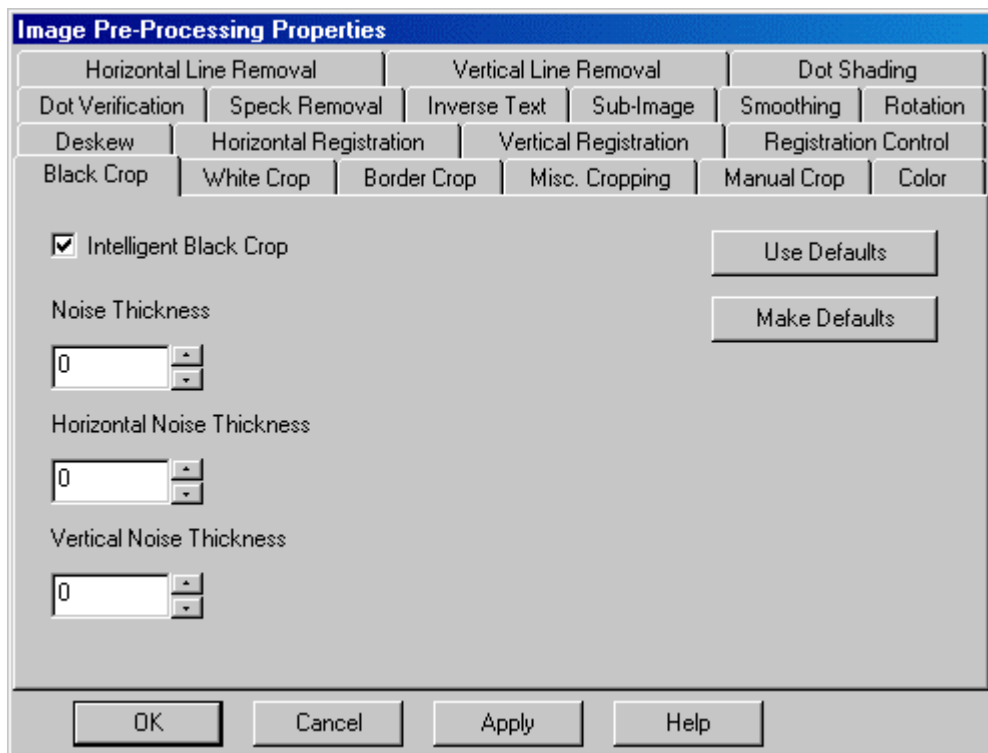
2. Select the **Intelligent Black Crop** check box to enable the Intelligent Black Crop fields on this tab. Black crop bleaches black margins to white.

3. In the **Noise Thickness** field, select the number of black pixels to consider as noise on the image. The range is 0 – 100 pixels.

4. In the **Horizontal Noise Thickness** field, select the number of black pixels of horizontal noise to bleach white in the image. Use this option to remove black noise on the left and right edges of the image. The range is 0 – 100 pixels.

5. In the **Vertical Noise Thickness** field, select the number of black pixels of vertical noise to bleach white in the image. Use this option to remove black noise on the top and bottom edges of the image. The range is 0 – 100 pixels.

6. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

7. Select **Apply** to activate your changes.

Setting White Crop Properties

White crop trims white space on the border of the image where no data displays. Use white crop only after you use the black crop feature to remove borders from images.

To set white crop properties:

1. On the Image Pre-Processing Properties dialog, select the White Crop tab.
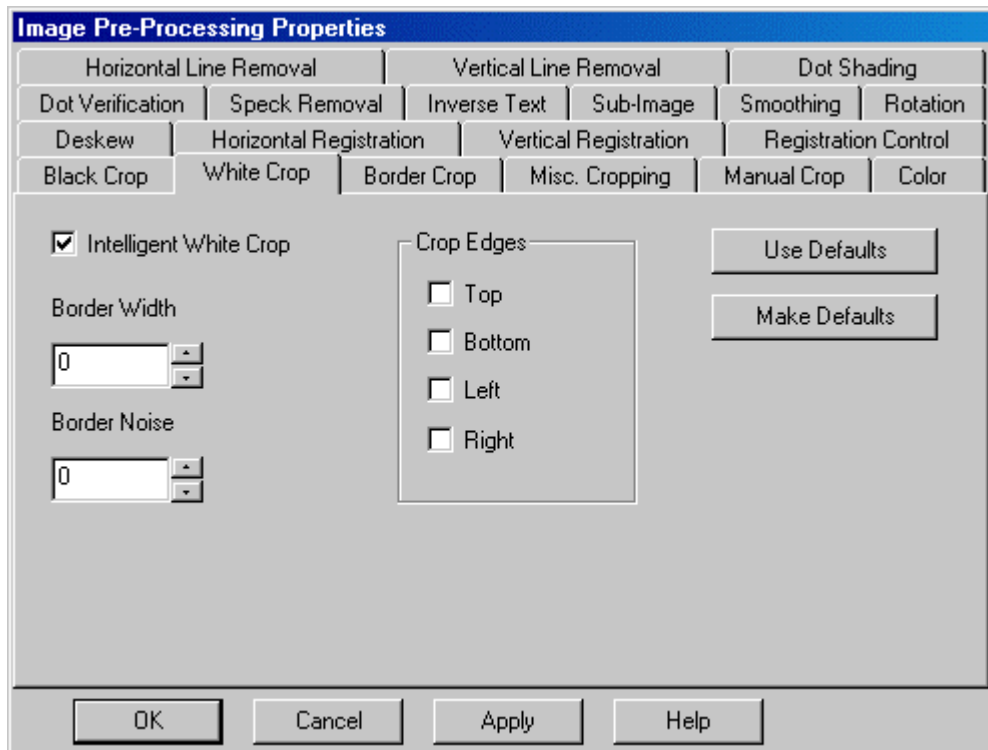
2. Select the **Intelligent White Crop** check box to enable the Intelligent White Crop fields on this tab.

3. In the **Border Width** field, select how wide the resultant border is after cropping. The range is 0 – 10,000 pixels.

4. In the **Border Noise** field, select how much border noise to ignore during the white cropping process. You can use this option to properly register an image that contains hand-written data in the border. The range is 0 – 100 pixels.

5. In the Crop Edges area, select which edges on the image to crop, according to the setting in the **Border Width** field. Options are **Top**, **Bottom**, **Left**, and **Right**.

6. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

7. Select **Apply** to activate your changes.

Setting Border Crop Properties

Border crop uses thresholds to determine areas on the image to bleach or crop. Measure all border crop settings from the outside edge of the image.

To set border crop properties:

1. On the Image Pre-Processing Properties dialog, select the Border Crop tab.
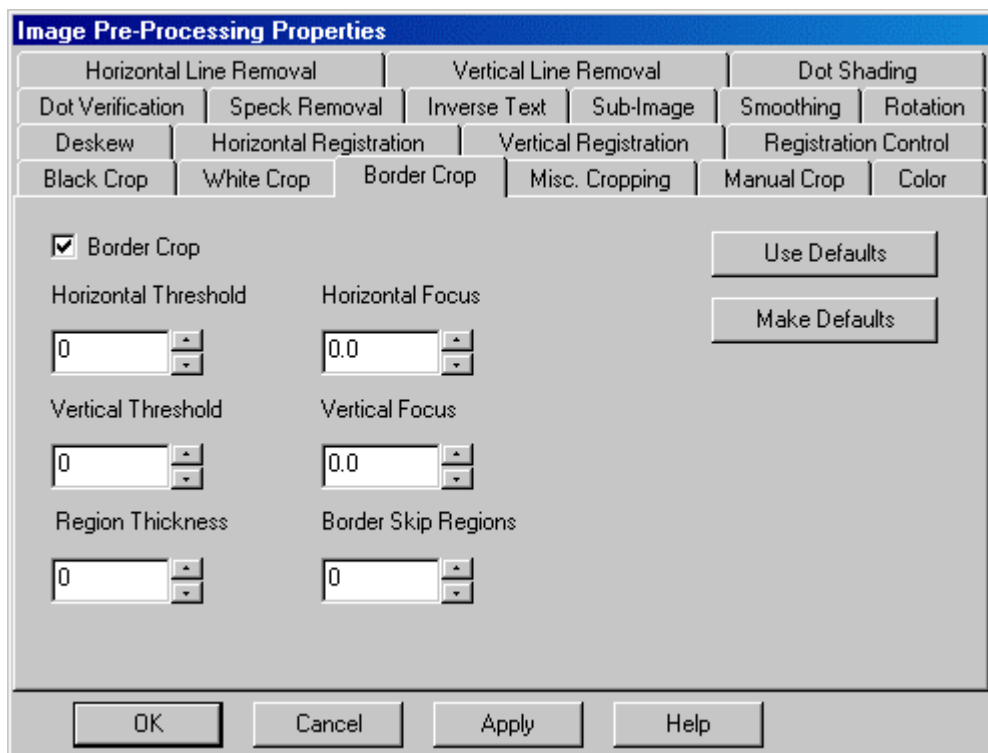
2. Select the **Border Crop** check box to enable the Border Crop fields on this tab.

3. In the **Horizontal Threshold** field, select the threshold density used for vertical strips of pixels. The first strip on the image edge that has a density lower than this threshold determines the image border. You must use this option with the **Region Thickness**. The range is 0 – 100 pixels.

4. In the **Vertical Threshold** field, select the threshold density used for horizontal strips of pixels. The first strip on the image edge that has a density lower than this threshold determines the image border. You must use this option with the **Region Thickness**. The range is 0 – 100 pixels.

5. In the **Region Thickness** field, select the pixel thickness of the strips. The range is 0 – 100 pixels.

6. In the **Horizontal Focus** field, select the number of parts into which each horizontal strip splits when calculating density. The range is 6,553.6 – 6563.6.

7. In the **Vertical Focus** field, select the number of parts into which each vertical strip splits when calculating density. The range is 6,553.6 – 6,563.6.

8. In the **Border Skip Regions** field, select the number of strips to ignore. Border determination begins after counting the specified number of strips. The range is 0 –10.

9. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

10. Select **Apply** to activate your changes.

Setting Miscellaneous Cropping Properties

Miscellaneous crop enhances your image by removing black objects and adding a rectangular border.

To set miscellaneous crop properties:

1. On the Image Pre-Processing Properties dialog, select the Misc. Cropping tab.
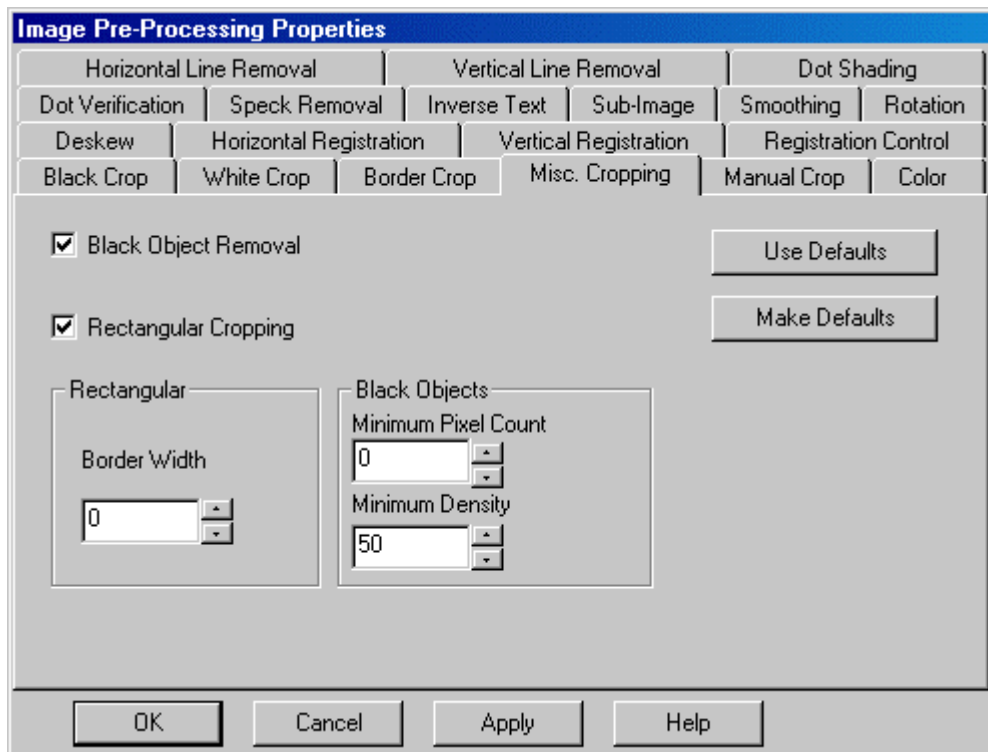
2. Select the **Black Object Removal** check box to enable the Black Objects settings on this tab. IPP removes black spots from the image based on the pixel count and density you specify.

3. Select the **Rectangular Cropping** check box to enable the Rectangular setting on this tab. Rectangular cropping deskews the image, crops it, and leaves a white border.

4. In the **Border Width** field, select the width of the white border to place around the image after rectangular cropping occurs if you selected the Rectangular Cropping check box. The range is 0 – 500 pixels.

5. In the **Minimum Pixel Count** field in the Black Objects area, select the minimum number of pixels to consider for black object removal. The range is 0 – 32,767 pixels.

6. In the **Minimum Density** field, select the minimum density to consider for black object removal. Density is the number of black pixels compared to the total number of pixels in the area. The range is 0 – 500 pixels.

7. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

8. Select **Apply** to activate your changes.

Setting Manual Crop Properties

Manual cropping resizes images by adding or removing pixels. Any value of zero on this tab disables the respective manual crop feature for that direction.

To set manual crop properties:

1. On the Image Pre-Processing Properties dialog, select the Manual Crop tab.
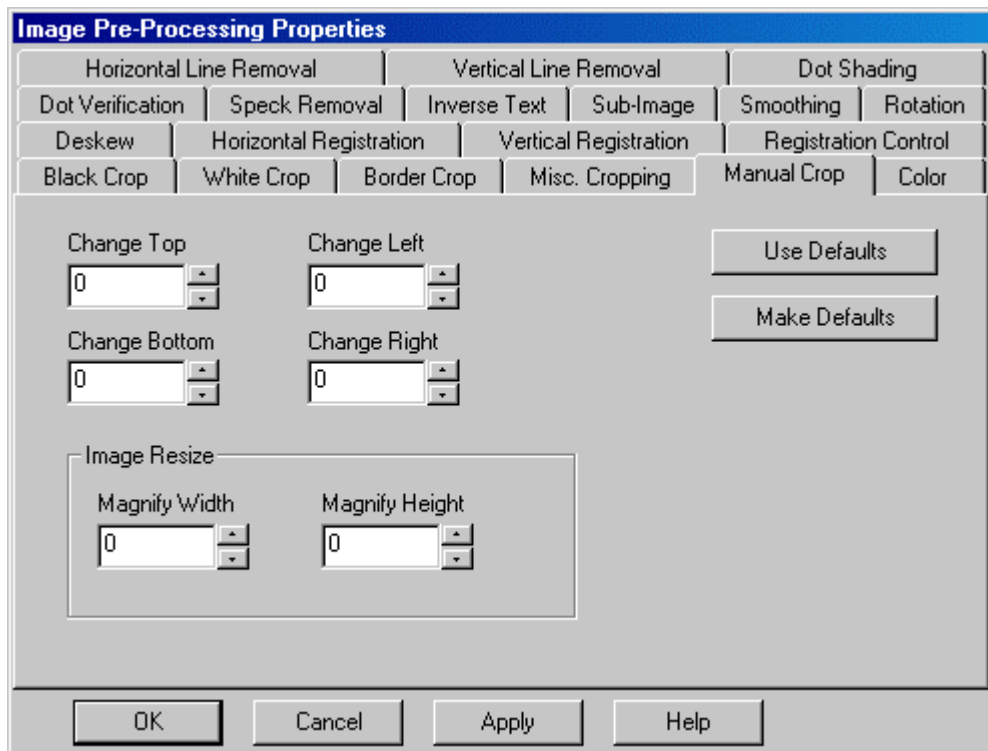
2. In the **Change Top** field, select the number of pixels to add or remove from the top of the image. In each of the Change fields, a positive value adds pixels, while a negative value removes pixels. The range for these fields is -10,000 – 10,000 pixels.

3. In the **Change Bottom** field, select the number of pixels to add or remove from the bottom of the image.

4. In the **Change Left** field, select the number of pixels to add or remove from the left side of the image.

5. In the **Change Right** field, select the number of pixels to add or remove from the right side of the image.

6. In the **Magnify Width** field, select the amount to enlarge or shrink the image width. This is the output width. This setting changes the image resolution. The range is 0 – 10,000 pixels.

7. In the **Magnify Height** field, select the amount to enlarge or shrink the image height. This is the output height. This setting changes the image resolution. The range is 0 – 10,000 pixels.

8. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

9. Select **Apply** to activate your changes.

Setting Dot Shading Properties

Dot shading removes individual dots, or dithering, from the image. Tables and borders are some of the areas that may use small dots to simulate shades of gray. Removing this shading improves image compression. In some cases, you can achieve a 50% or more compression rate by removing these dots.

To set dot shading properties:

1. On the Image Pre-Processing Properties dialog, select the Dot Shading tab.
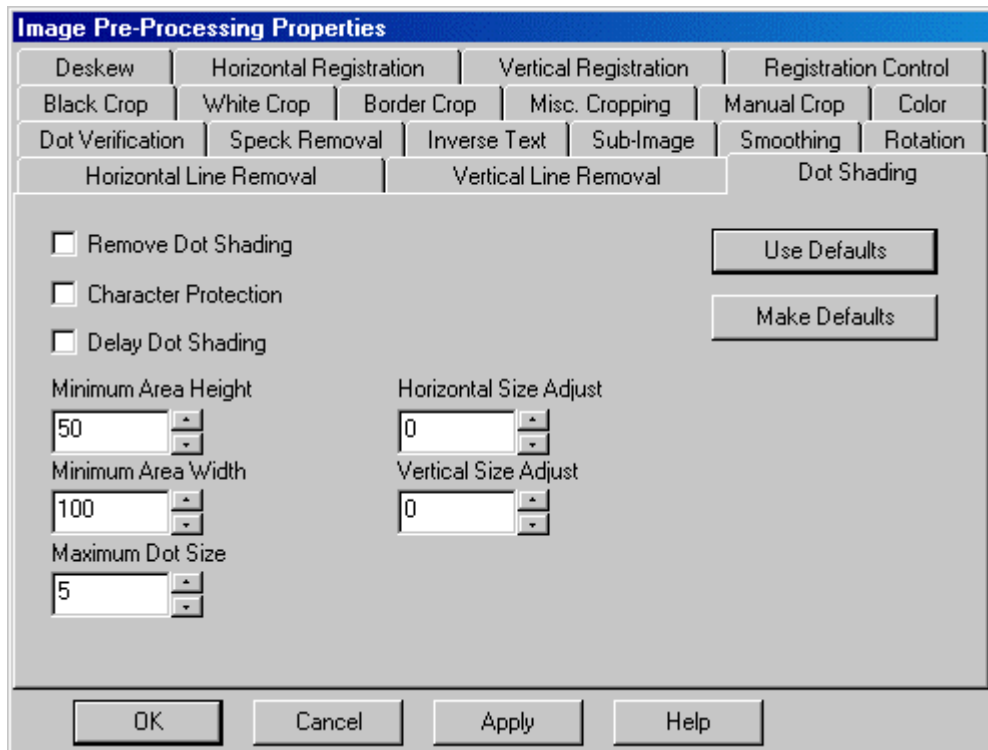
2. Select the **Remove Dot Shading** check box to enable the dot shading fields on this tab. Dot shading reduces the size of your image files by removing the small dots used to simulate shades of gray.

3. Select the **Character Protection** check box to prevent degrading text characters on the image during the dot removal process.

4. Select the **Delay Dot Shading** check box to perform the dot removal process after the line removal and image despeckling processes.

5. In the **Minimum Area Height** field, select the minimum height of dot-shaded zones to process. The default value is 50 and the range is 1 – 10,000 pixels.

6. In the **Minimum Area Width** field, select the minimum width of dot-shaded zones to process. The default value is 100 and the range is 1 – 10,000 pixels.

7. In the **Maximum Dot Size** field, select the maximum size of the dots to remove. The process removes all dots up to and including this size from the image. The default value is 5 and the range is 1 – 20 pixels.

8. In the **Horizontal Size Adjust** field, select the number of pixels to refine the **Maximum Dot Size** to remove dots that are wider than the limit. Setting this value to 1 removes all dots that are one pixel wider than the **Maximum Dot Size** setting. Set this value to a negative number if IPP is degrading text. The range is -10 – 10 pixels.

9. In the **Vertical Size Adjust** field, select the number of pixels to refine the **Maximum Dot Size** limit to remove dots that are taller than the limit. Setting this value to 1 removes all dots that are one pixel taller than the **Maximum Dot Size** setting. Set this value to a negative number if IPP is degrading text. The range is -10 – 10 pixels.

10. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

11. Select **Apply** to activate your changes.

Setting Dot Verification Properties

Use the options on the Dot Verification tab to further refine the Dot Shading settings.

To set dot verification properties:

1. On the Image Pre-Processing Properties dialog, select the Dot Verification tab.
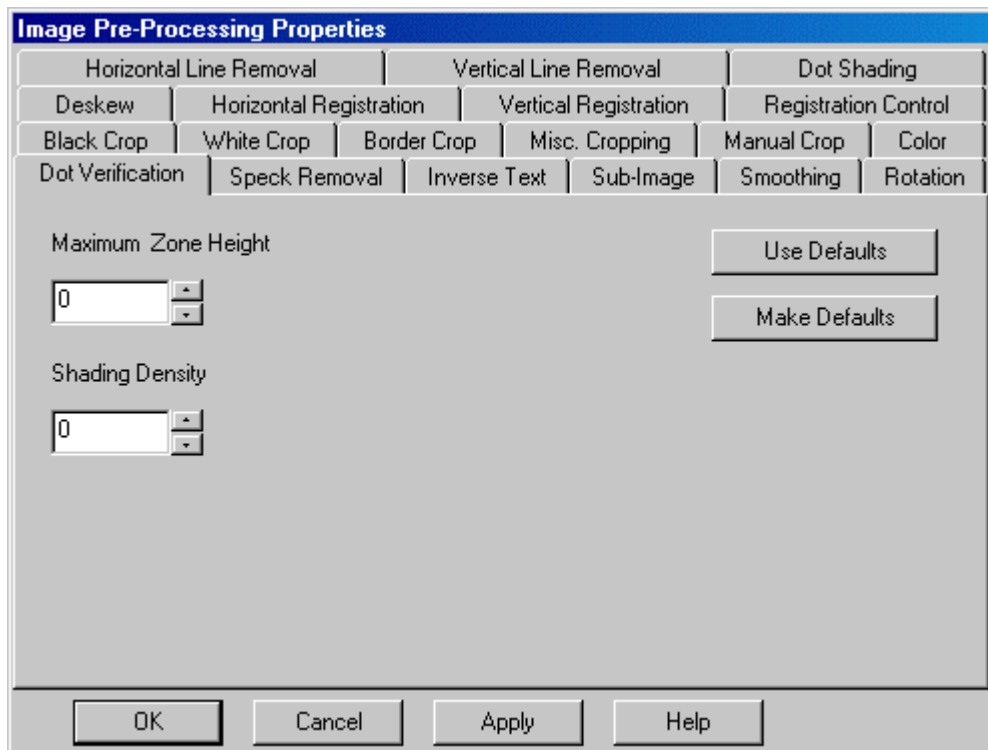
2. In the **Maximum Zone Height** field, select the upper limit on the size of dot-shaded regions where dot count analysis is performed. This value refers to the number of scan lines on the image. The lines are measured in pixels. During the dot shading process, no analysis occurs within a zone that has more scan lines than the number in this field. The range is 0 – 500 pixels.

3.   In the **Shading Density** field, select a value to adjust the detection algorithm for shaded zones. For areas that are sparsely shaded, type a small negative number. As the density increases, increase the number. Density is the amount of black pixels compared to the total number of pixels in an area. The range is -100 – 100 pixels.

4.   To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

5.   Select **Apply** to activate your changes.

Setting Speck Removal Properties

Speck removal (despeck) removes isolated black marks from the image. This reduces the amount of noise on images. Horizontal despeck runs before vertical despeck.

To set speck removal properties:

1.   On the Image Pre-Processing Properties dialog, select the Speck Removal tab.
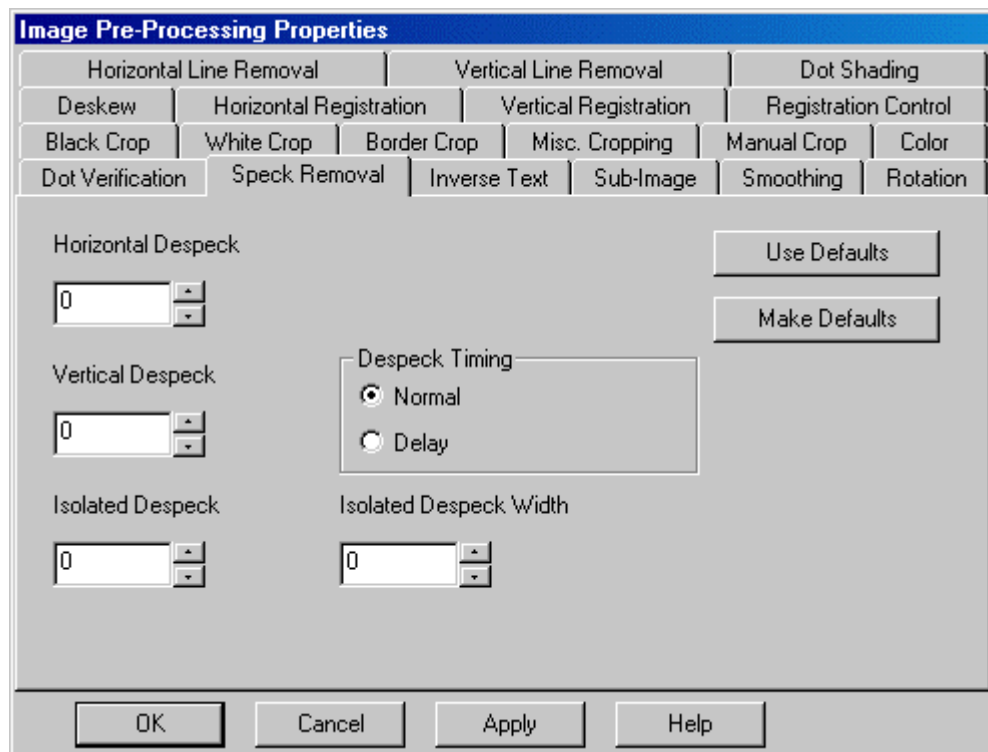
2.   In the **Horizontal Despeck** field, select the maximum horizontal size of specks to remove from the image. Increase this value to remove larger specks or decrease it to prevent the removal of small letters from the image. Horizontal despeck is faster than vertical despeck. The range is -100 – 100 pixels.

3.   In the **Vertical Despeck** field, select the maximum vertical size of specks to remove from the image. The range is -100 – 100 pixels.

4.   In the **Isolated Despeck** field, select the size of specks to remove that are not touching anything else on the image. Setting this value too high will remove dots on text characters, such as i or j. The range is -100 – 100 pixels. A negative value activates the diagonal protection feature. Use this if despeck is removing diagonal portions of characters. Isolated despeck is faster than horizontal or vertical despeck.

5. In the **Isolated Despeck Width** field, select the width of the isolated despeck feature. Use this field with Isolated Despeck. A negative number activates the isolated despeck height or width algorithm that enables searches for speck height or width. The range is -100 – 100 pixels.

6. In the Despeck Timing area, select the value that determines when the despeck process occurs. Options are:
   - o **Normal** - Performs the despeck function early within IPP. This is the default.
   - o **Delay** - Performs the despeck function late within IPP after removal through isolated despeck and dot shading.

7. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

8. Select **Apply** to activate your changes.

Setting Inverse Text Properties

Inverse text settings convert white text on a black background to black text on a white background. Recognition engines cannot read inverse text; therefore, use this option on any inverse text image.

Note: The image must be deskewed either prior to, or at the same time as, the inverse text processing. If the image is not deskewed, then results may not be reliable for detection and inverse of inverse text areas on the image. No other settings are required to be enabled except for the Text Inversion check box.

To set inverse text properties:

1. On the Image Pre-Processing Properties dialog, select the Inverse Text tab.
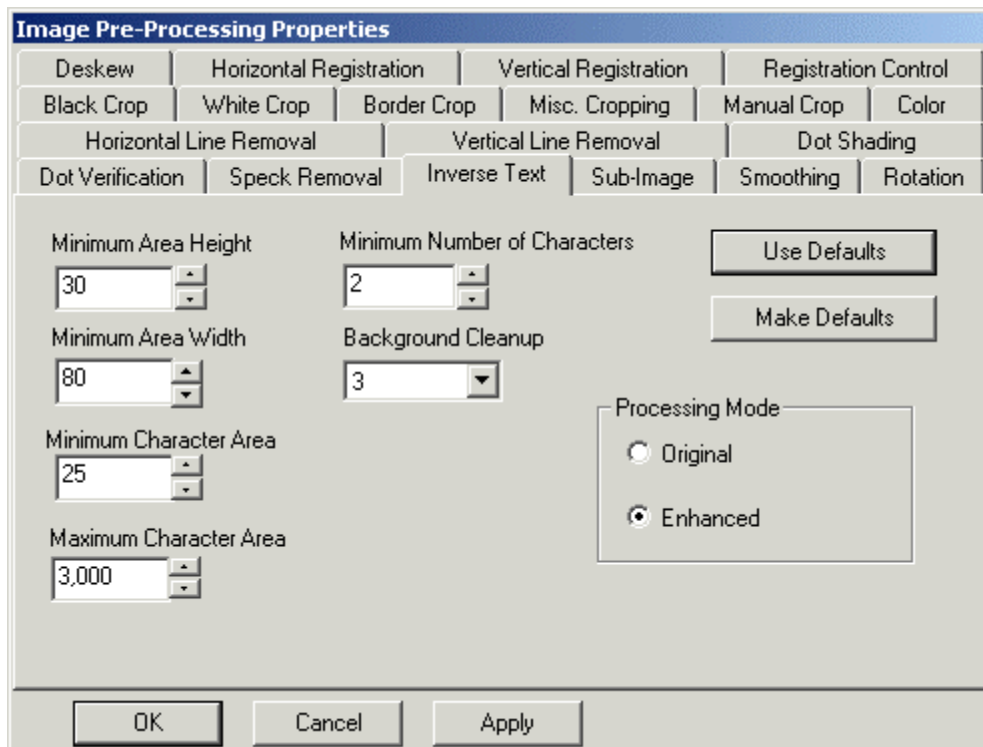
2. In the **Minimum Area Height** field, select the minimum height of inverse text areas to process. The default value is 30 and the range is 1 – 10,000 pixels. If this is set too low, then dense data, such as multi-line small print, might be detected as inverse text. If it is set too high, it may miss detecting inverse text.

3. In the **Minimum Area Width** field, select the minimum width of inverse text areas to process. The default value is 80 and the range is 1 – 10,000 pixels.

4. In the **Minimum Character Area** field, select the minimum character area (white pixels) for each character within an inverse text area. The default is 25 pixels and the range is 1 – 10,000 pixels. If this is set too small, then items such as large bold characters with a "hole", such as the letter "P", might be treated as inverse text. If it is set too high, it may miss detecting inverse text.

5. In the **Maximum Character Area** field, select the maximum character area (white pixels) for each character within an inverse text area. The default is 3000 pixels and the range is 1 – 10,000 pixels. If this is set too small, then large point size inverse text may fail to be processed.

6. In the **Minimum Number of Characters** field, select the minimum number of characters within an inverse text area. The default is 2 and the range is 1 – 100 characters. If it is set too low, then items such as large bold characters with a "hole", such as the letter "P", might be treated as inverse text. If it is set too high, it may miss detecting inverse text.

7. In the **Background Cleanup** field, select the size of specks in the image to remove prior to detection of the inverse text regions. The default is 3 pixels (3x3 filter). You can select from the following pixel filters: 3 pixels (3x3), 5 pixels (5x5), 7 pixels (7x7), or 9 pixels (9x9).

8. In the Processing Mode area, select the option that specifies how you want to process inverse text:

   • **Original** - 4.6 and earlier implementation with different fields (see the documentation for the earlier release).

   • **Enhanced** - 5.0 implementation

9. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

10. Select **Apply** to activate your changes.

Setting Sub-Image Properties

Sub-image functions extract a specified processing area from images, such as an address block on a form. The extracted sub-image overwrites the original image on the disk. By using this feature, you can produce and save smaller image files that load into memory faster and require less recognition processing time.

Note: Run this process after running deskew and image registration, but prior to other processes.

To set sub-image properties:

1. On the Image Pre-Processing Properties dialog, select the Sub-Image tab.
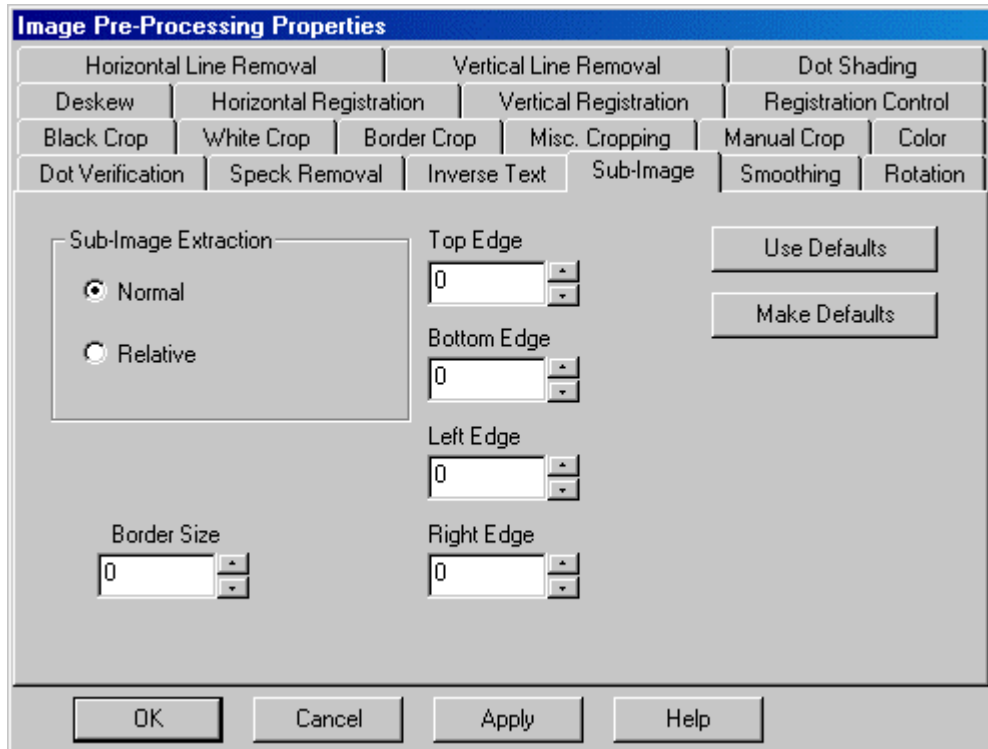
2. In the Sub-Image Extraction area, select the value that determines. Options are:

    o   **Normal** - Use the values in the **Edge** fields as absolute values for sub-image
        extraction. The edge settings determine the absolute size of the sub-image you want
        to extract. Normal values for sub-image extracting can be positive numbers only.

    o   **Relative** - Use the values in the **Edge** fields as relative adjustment values for sub-
        image extraction. Positive numbers expand, while negative numbers remove pixels
        from the edge. Using this feature enables negative values for settings.

3. In the **Border Size** field, select the size of a white border around the newly extracted sub-
    image. The border will be the width that you specify in this field. This field is only active
    when you select Normal extraction. The range is 0 – 10,000 pixels.

4. In the **Top Edge** field, select the value that defines the top edge of the sub-image.
    Measurements start from the top of the image down. A value of 0 is the very top of image.
    The range is -10,000 – 10,000 pixels.

5. In the **Bottom Edge** field, select the value that defines the bottom edge of the sub-image.
    Measurements start from the top of the image down. A value of 0 is the very bottom of the
    image. The range is -10,000 – 10,000 pixels.

6. In the **Left Edge** field, select the value that defines the left edge of the sub-image.
    Measurements start from the left edge of the image across to the right. A value of 0 is the left
    edge of the image. The range is -10,000 – 10,000 pixels.

7. In the **Right Edge** field, select the value that defines the right edge of the sub-image.
    Measurements start from the left to the right edge of the image. A value of 0 is the right edge
    of the image. The range is -10,000 – 10,000 pixels.

8. To cancel your changes and return to the default settings, select the **Use Defaults** button. To
    make your changes the new defaults, select the **Make Defaults** button.

9. Select **Apply** to activate your changes.

Setting Smoothing Properties

The Smoothing feature fills small pits and removes small bumps, or hairs, on the surface of characters to reduce the image size and improve appearance. If you set options in the **Customize** area, use the **Add**, **Change**, or **Delete** buttons to apply your customized growth or erosion factor to the image.

Note: **Change** and **Delete** work on the last item in the custom box. You can create a maximum of four custom settings for each job.

To set Smoothing properties:

1. On the Image Pre-Processing Properties dialog, select the Smoothing tab.
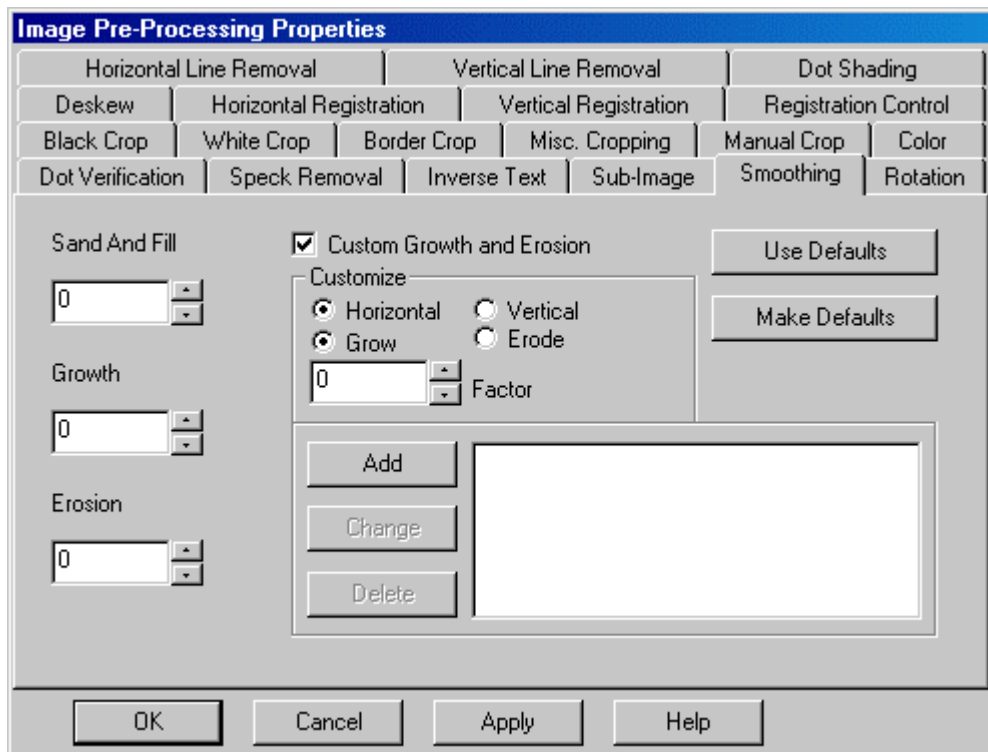
2. In the **Sand and Fill** field, select the size of pits and bumps on characters to fill or sand. A larger numeric value in this field repairs larger pits and bumps. Settings between 1-5 work well. The range is 0 – 100 pixels.

3. In the **Growth** field, select the amount to increase the number of black pixels on the image. For example, a growth value of two causes a single pixel to become a 3x3 pixel. Use this option to connect dot matrix or darken light print. Growth is applied before Erosion. The range is -100 – 100.

4. In the **Erosion** field, select the amount to decrease black pixels on the image. Use this option to thin bold print. The range is 0 – 100 pixels.

5. Select the **Custom Growth and Erosion** check box to enable the custom growth and erosion controls on this tab. You can add a maximum of four customizations per job.

6. In the Customize area, select a combination of values to customize the growth and erosion smoothing on your image. To do this:

   a. Select **Horizontal** or **Vertical** and **Grow** or **Erode**. Select the combination of values that will sufficiently increase or decrease the number of black pixels on your image.

     b. In the **Factor** field, select the amount to increase or decrease black pixels on the image. The range is 0 – 100 pixels.

     c. Select the **Add** button to add this custom setting.

     d. To modify a setting, highlight it, change the value in the **Factor** field and select the **Change** button.

     e. To remove a setting, highlight it and select the **Delete** button.

7. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

8. Select **Apply** to activate your changes.

Setting Registration Control Properties

Registration control provides much finer control over the image registration functions. When you select horizontal or vertical registration, the software checks the edges of the image for consistent activity. Text in the image contains a high level of consistent activity, while edge noise contains a low level of activity. White space has no activity at all. The registration control settings allow you to define the sensitivity level of the software to this activity and adjust image registration accordingly.

To set registration control properties:

1. On the Image Pre-Processing Properties dialog, select the Registration Control tab.
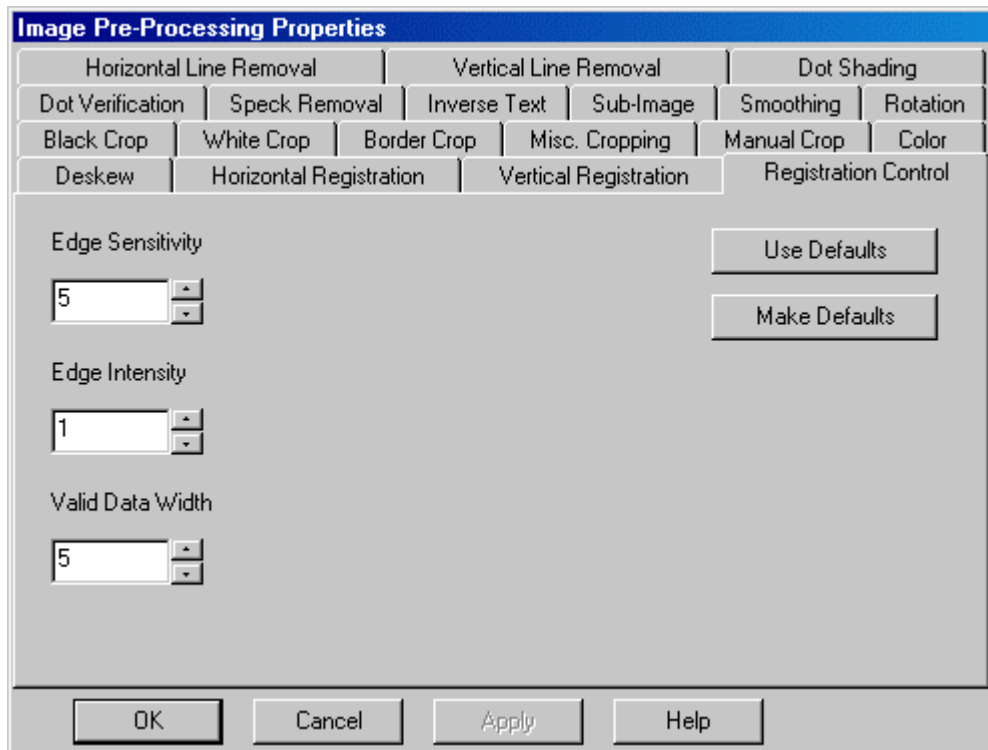
2. In the **Edge Sensitivity** field, select the value that determines the software sensitivity to white space on the edge of the image. This setting instructs the IPP process to look at the white space between the black pixels to determine the density and consistency of the image edge noise. Setting too high a value can cause valid data to be treated as edge noise. Setting too low a value can cause edge noise to be considered as valid data. The range is 0 – 500 pixels.

3. In the **Edge Intensity** field, select the value that determines how intense the level of activity must be on the image for it to be considered valid. Text is a high level of consistent activity, edge noise is low, and white space has none. Setting this option to a value other than 1 can improve image registration if, in a previous registration attempt, your image registered on edge noise. The range is 0 – 500 pixels.

4. In the **Valid Data Width** field, select the width of valid data on the image. Use this option to ignore perpendicular lines on the image. However, if your images contain dot matrix print, you may need to lower this value to 1. The range is 0 – 500 pixels.

5. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

6. Select **Apply** to activate your changes.

Setting Horizontal Line Removal Properties

Removing horizontal lines from the image is important if you are performing recognition on the image. Horizontal line removal can improve recognition accuracy by eliminating certain horizontal lines from the image, especially lines that are extremely close to the text you want recognized.

To set horizontal line removal properties:

1. On the Image Pre-Processing Properties dialog, select the Horizontal Line Removal tab.

2. In the **Minimum Line Length** field, select the minimum length of horizontal lines to remove from the image. Set this value so that the process overlooks portions of characters like T or L. The range is 1 – 10,000 pixels.

3. In the **Maximum Line Width** field, select the maximum thickness of horizontal lines to remove from the image. Using this setting can prevent removal of headlines or large titles from the image. The range is 1 – 1,000 pixels.

4. In the **Maximum Line Gap** field, select the largest gap allowed in a horizontal line. The process fills gaps that fall within this value with black pixels and then removes the line. The range is 1 – 1,000 pixels.

5. In the **Edge Cleaning Factor** field, select the number of pixels to erase on either side of the removed horizontal line. This setting enables you to remove broken or poor-quality lines. The default value is 2. Set this value higher if a portion of a line remains after you run the line removal process. Set the value lower if the process is removing any text. The range is 1 – 100 pixels.

6. In the **Reconstruction Width** field, select the maximum width of characters to repair. The default is 20 pixels. Set this higher if the process is not repairing text on the image or lower if the reconstructed text has errors. Set this value to zero to disable width reconstruction. The range is 0 – 1,000 pixels.

7. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

8. Select **Apply** to activate your changes.

Setting Vertical Line Removal Properties

Vertical line removal removes vertical lines from the image. This is particularly important if you are performing recognition on the image. Vertical line removal can improve the recognition accuracy rate by eliminating certain vertical lines from the image, especially lines that are extremely close to the text that you want recognized during the recognition process.

To set vertical line removal properties:

1. On the Image Pre-Processing Properties dialog, select the Vertical Line Removal tab.
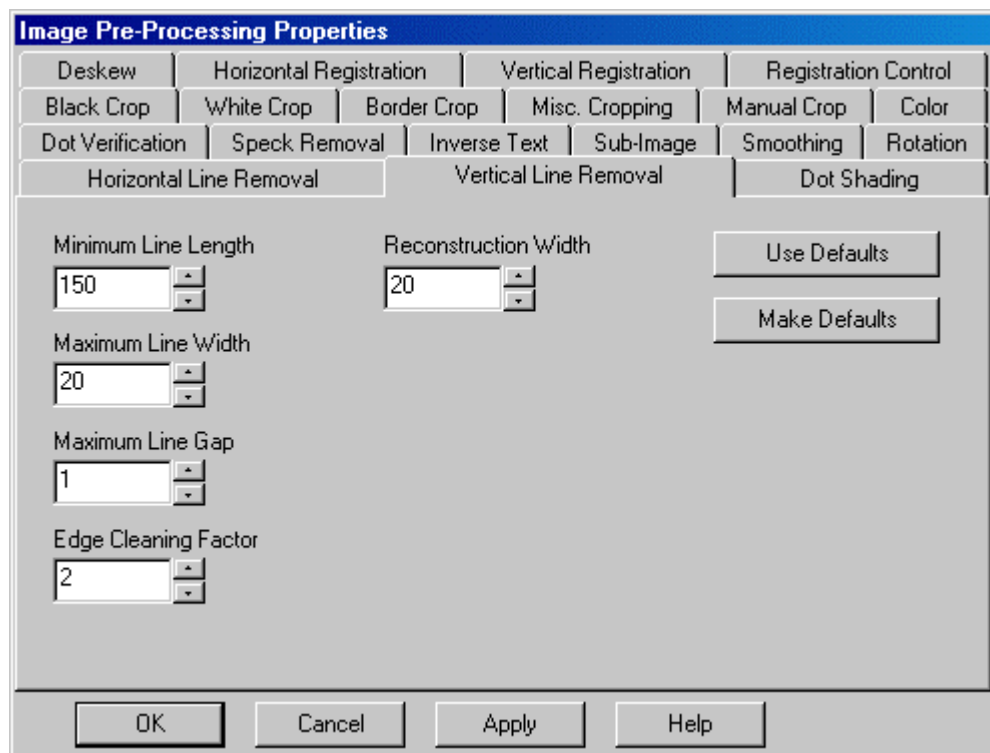
2. In the **Minimum Line Length** field, select the minimum length of vertical lines to remove from the image. You must set this value larger than vertical portions of any text or the process may remove parts of characters like L and T. The range is 1 – 300 pixels.

3. In the **Maximum Line Width** field, select the maximum thickness of vertical lines to remove from the image. Setting this value can prevent removal of characters in headlines or large titles. The range is 1 – 1,000 pixels.

4. In the **Maximum Line Gap** field, select the largest gap allowed in a vertical line. If a gap falls within this value, the process inserts black pixels to fill it in, and then removes the reconstructed line. The range is 1 – 1,000 pixels.

5. In the **Edge Cleaning Factor** field, select the number of pixels to erase on either side of the removed vertical line. This enables you to remove broken or poor-quality lines. The default value is 2. Set this value higher if a portion of a line remains after you run the line removal process. Set the value lower if it is removing any text. The range is 0 – 1,000 pixels.

6. In the **Reconstruction Width** field, select the maximum width of characters to repair. The default is 20 pixels. Set this value higher if the process is not repairing text on the image or lower if the reconstructed text has errors. Set this value to zero to disable this process. The range is 0 – 1,000 pixels.

7. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

8. Select **Apply** to activate your changes.

Setting Color Properties

The Designer module supports using color JPG images. You can also use a color image as your setup image. During a process prior to IPP, a FW_IMT=COLOR tag must be inserted in the BDF before the names of color images that do not have a .JPG extension. If the BDF contains level tags and all images within a given level are color images, then only one tag is written at the beginning of each level to identify all images within that level as color images.

When you run IPP against a color setup image, the IPP process creates a corresponding black and white TIF image. When the black/white image is not successfully created during this process, no error message displays and Designer launches the Image Viewer with a single color image in it.

The options on the Color tab determine how color images are processed during Color IPP at the job level. Color IPP processing is not supported at the form level. All color processing is performed within the job level using the IPPClnt program prior to the typical binary image IPP processing.

To set color properties:

1. On the Image Pre-Processing Properties dialog, select the Color tab.
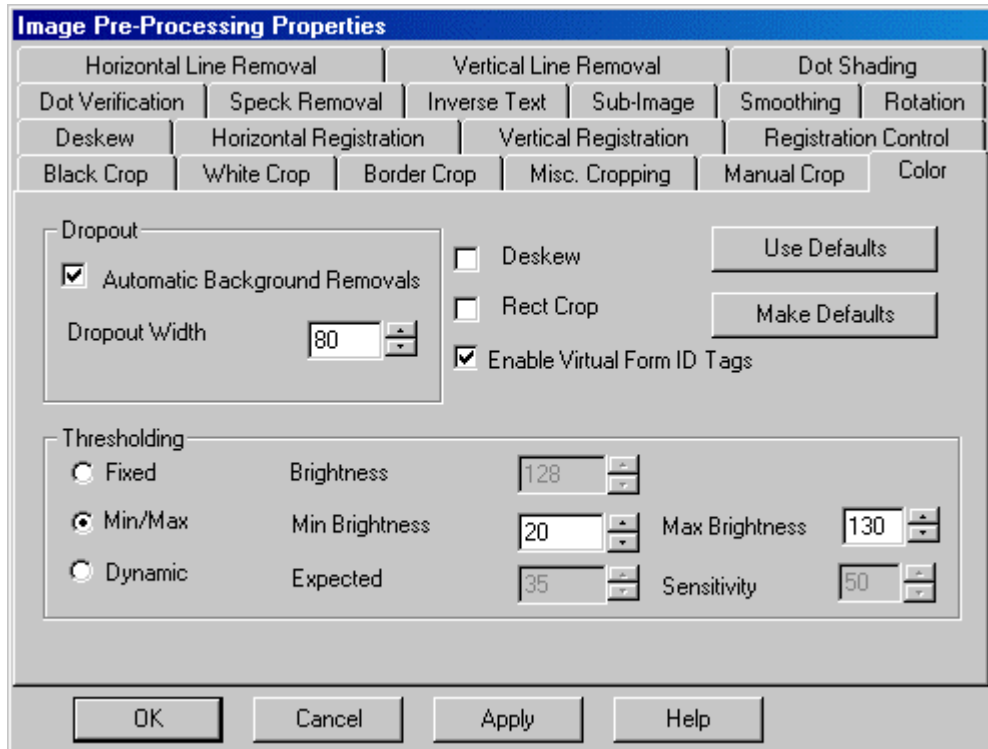
2. Select the **Automatic Background Removal** check box to remove the background from the input color image prior to applying threshold settings to a bi-tonal image. Improved OCR on the resultant bi-tonal image is likely for images that have either dark blue backgrounds or colored text on a similarly colored background. The output color image is unaffected by this setting.

3. In the **Dropout Width** field, select the width in "brightness" around the background color to be used when dropping out the background color prior to the image threshold. The higher this value, the more background noise and variations are removed. A width of "0" means that only the "average" background value itself will be removed and no allowances are made for variation. Increase this value to have more of the background removed. The default recommended width is 40. The maximum width allowed is 255.

4. Select the **Deskew** check box to enable color deskew and overwrite the original input color image with the deskewed image. The output bi-tonal image preserves the same deskew.

5. Select the **Rect Crop** check box to enable border cropping and overwrite the original input color image with the cropped image. The output bi-tonal image preserves the same cropping. This causes the output bi-tonal image to be written out with no black borders (smaller image). If cropping is turned off, the black borders are preserved, leaving the image the same size as the input image.

6. Select the **Enable Virtual Form ID tags** check box to enable writing out the background color value (BackVal=) to the BDF. The BackVal parameter is a string that represents the automatically-detected background color of the image The Form ID process uses this tag to identify an image based on the color of the background.

7. In the Thresholding area, select from the following options to define how thresholds are determined:

   - **Fixed** - Use the fixed (global) brightness threshold for the entire image. This is the default. Set the value in the **Brightness** field to the fixed brightness threshold value in the

range 0–255. This is the simplest threshold method and works best when all of the images in the job have similar background brightness and text darkness. The default fixed brightness threshold is 127.

- **Minimum/ Maximum** - Automatically select the best global threshold for the entire image within the range you define in the **Minimum Brightness** and **Maximum Brightness** fields. This is the suggested threshold method to use when images in a given job vary in contrast and color between background and text. This is best for applications that have the most variation from form to form.

- **Dynamic -** Use an adaptable threshold that automatically sets a different threshold pixel by pixel. In the **Expected** field, select the expected threshold value (the default is 127). In the **Sensitivity** field, select a range (the default is 50). Higher values for sensitivity enhance dynamic threshold in case of small or faint print, but sensitivities too large will cause noise in the resulting image. This is the suggested threshold method to use for applications that contain images that vary in contrast between the text and the background depending on where the location of the text is on a page (or where there are texts of different darkness on a single image). It is not suggested if there are different form types in a given job that vary in greatly in contrast from form to form.

8. To cancel your changes and return to the default settings, select the **Use Defaults** button. To make your changes the new defaults, select the **Make Defaults** button.

9. Select **Apply** to activate your changes.

Setting Form-Level IPP Properties

Form-level settings for IPP are applied in addition to, and after, any job-level IPP settings. For example, if you have multiple forms in your job that were scanned in sideways, you can perform IPP at the job-level to rotate all of the forms and then perform IPP at the form-level to remove large borders from each image.

Form-level IPP occurs as part of the Enhancement process. It is combined with form registration and zone removal results. Within Enhancement, IPP occurs in the following sequence:

- Form-specific Registration

- Form-specific IPP - Production images are enhanced after form identification to improve recognition results. Enhancement includes the removal of non-data related features of the form (i.e., form removal). Form removal reduces the size of the image and reduces the recognition processing time.

To set form-level IPP properties:

1. From Designer, double-click on the form to display the Form Properties dialog.

2. Select the IPP tab.

3. Select the **Remove Background** check box to remove all data that is outside of the recognition zones on the current data form. This can amount to removing a significant amount of data on the image, thereby reducing the image file size.

4. Select the **Use setup image as form removal template** check box only if you are using Form Removal within your application. See the Setting Form Removal Properties section for more information.

Note: Form removal using templates is not available for virtual forms. Therefore, this check box is disabled if a (variant) database exists for the current form.

5. Select the **Drop Out Ink** check box to remove features printed in non-scanning, drop out ink from a paper form during the scanning process. This type of forms processing affects recognition and mark sense applications. A setup image scanned to show all lines on a form is used for the design process. Printing forms in a color that the scan bulb does not see makes non-data features on the images invisible after the scanning process. This minimizes the size of images by only providing the recognition data on the form. You should scan setup images with a bulb so that the background is visible.

6. Select the **Overwrite** check box to replace the original production image with the image after enhancements. Applying IPP and Enhancement to a production image creates a second, *enhanced*, image. When you select this option, these combined results overwrite the original image. If you do not select the **Overwrite** checkbox, the results are output to the \FormWare\Images\<imagepath>.enh directory.

7. Select the **Enhance before recognition test** check box to run default Enhancement features regardless of whether you enable the Enhancement process. This allows you to force the Enhancement process on the current images when testing from within Designer.

8. In the Setup and Test area, select the **IPP** option to enable the **Properties** button that displays the IPP Properties dialog.

9. Select the job-level properties to be applied during form-level IPP. See the discussion of the Image Pre-Processing Options and Image Pre-Processing Properties dialogs in the Setting Job-Level IPP Properties section.

Note: The options and properties that do not apply to the form-level are grayed out on this dialog (e.g., Color Processing, Convert Image DPI (Fax), and Convert Intel to Group 4). Also, all tabs that are unsupported at the form level have been removed from the IPP Properties dialog.

Note: Any "registration type" settings (e.g., image registration, image rotation, or sub-image can potentially modify the location of data and lines on the image. Accordingly, it is suggested that you use the form-level IPP-overwritten image as the new setup image for the form if you have enabled (checked) any of these settings. You can do this by running enhancement on the current setup image (via the Image tab Enhance button) and "saving" the new output image as the new setup image.

---

10. In the Setup and Test area, select the **Registration** option to enable the Registration testing buttons.

11. Select the **Test** button to test Registration on a BDF for the current form. This test launches the Enhancement process in "Registration Statistics" mode. The results display in the Batch Test Review dialog. Use a new BDF for Registration testing. During the test, command lines are written into this BDF render it unusable for any other testing or processing purposes. This button is active only when you select the **Registration** option.

12. Select the **Review** button to display the Batch Test Review dialog with the image and its registration zones highlighted in red. Up to four registration zones at a time are displayed on the left side of the dialog. Within any of these windows, right-click to display the Registration Anchor dialog. This enables you to manually edit a field's registration parameters. View up to twelve registration zones for any given image using the directional arrows. This button is active only when you select the **Registration** option. See the Testing Image Registration section for information about the Batch Test Review dialog.

13. Select the **Auto Tune** button to run Enhancement in "Form Registration Tuning" mode. Auto Tune tries to automatically adjust registration parameters. This button is active only when you select the **Registration** option.

---

14. In the **BDF** field, select the **Browse** button to select a BDF for Registration testing. This button is only enabled when you select the **Registration** option. Select the **Remove** button to remove the BDF name from the field or select the **Edit** button to edit the selected BDF.

15. Select the **Auto Remove Background** button to locate and create zone-level removal for horizontal and vertical lines on the form. Zone-level removal removes lines in a recognition zone and can also apply image filters for removing image noise and for smoothing dot matrix print. This process is equivalent to displaying the Removal Features dialog (from the Field Properties, IPP tab, **Enhancement** button) and selecting the **AutoFind** button, but at form level, rather than an individual field level (i.e., the process searches the entire form, not just one registration field zone).

16. Select the **Clear Auto Removals** button to deletes all removal zones created on the form during the Auto Remove process. A confirmation message displays to verify that you want to delete the existing removal settings. Select **Yes** to continue deleting the removals.

Setting Field-Level IPP Properties

In addition to job-level and form-level IPP settings, you can also set IPP properties at the field level. You use the options on the Field Properties, IPP tab to select zone-level removal options for fields on your data form. This process occurs after form-level IPP decreasing the need for extensive field level removal.

To set field-level IPP properties:

1. Double-click the field on the form for which you want to set IPP properties. The Field Properties dialog displays.

2. Select the IPP tab.
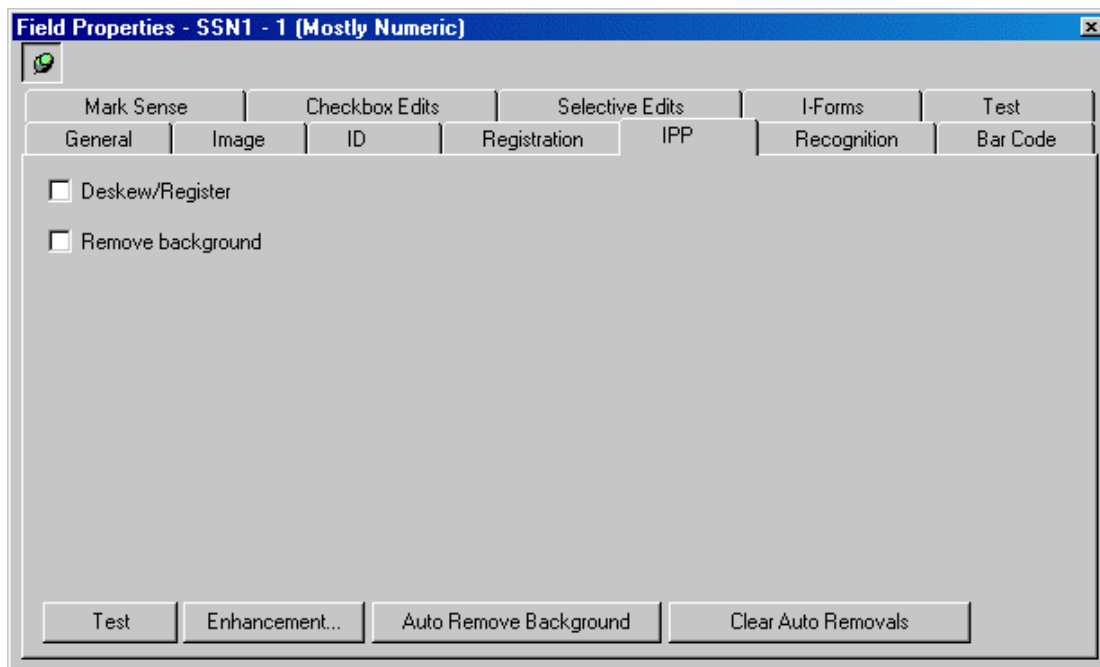
3. Select the **Deskew/Register** check box to perform deskew and registration functions on the current field. Use this for fields such as labels that may have a different skew than the rest of the form page.

4. Select the **Remove background** checkbox to remove everything on the image that is not contained in a recognition zone. This type of removal can improve recognition processing because the image after removal is smaller and processes more quickly.

5. Select the **Test** button to test the image enhancement options on the current field.

6. Select the **Enhancement** button to display the Removal Features dialog where you set zone-level removal for the field. See the Creating Recognition Zones section for more information about this task.

7. Select the **Auto Remove Background** button to perform the Auto Remove process on the current field. This process locates and creates removal zones for horizontal and vertical lines on the form background. This is equivalent to selecting the Enhancement button and then selecting the **AutoFind** button.

8. Select the **Clear Auto Removals** button to delete all removal zones created in the field during the Auto Remove process. A confirmation message displays to verify that you want to delete the existing removal settings. Select **Yes** to delete the removal settings at the field-level.

9. Select **Save All** to save your IPP settings.

Testing IPP Settings

There are two IPP test levels: the form-level and the job-level. You can perform both discrete and comprehensive tests on your IPP settings. Discrete testing tests just the IPP settings for the job (including the form settings). Comprehensive testing tests the IPP settings along with other tasks and against either a batch of images or a single image.

Note: When conducting a comprehensive test, the BDF must contain a JOBNAME=<jobname> parameter, where <jobname> is the name of the current job. Without this variable, the test will fail.

Testing IPP Against the Setup Image

Now that you have set and selected the IPP options and properties you want to use, you should test these options on a data form within the job. This test will run form-level IPP settings on the setup image. Follow these steps to perform a discrete test of IPP properties against your setup image:

1. Open a job and form within Designer and display the Form Properties, Image tab.

2. Select the **IPP** button to start the test.

3. The test results display in the image viewer.

4. With your cursor over the image, right-click and hold your mouse button to move the image within the window. You can view the IPP changes to the setup image.

5. Select **OK** from the bottom of the window to accept the enhanced image, or select **Cancel** to reject it. Accepting the changes overwrites your setup image file (SUP) with the processed image.

Note: If you have selected the **View Tests in Completion** button on the Designer window for the current job, only the processed image displays within a Completion window. Therefore, for the purposes of this test, you should deselect this option so that you can compare the original image to the processed image within Designer.

Testing IPP Against a Batch

You can test IPP settings on a batch using the Job Properties dialog, Test tab. Testing IPP alone is a transparent operation. Job-level IPP settings are applied to each image in the test BDF. At the completion of the test, the modified images overwrite the tested images stored in the \FormWare\Images folder.

Follow these steps to test IPP job properties on a batch:

1. Open a job within Designer.

2. Display the Job Properties, Test tab.
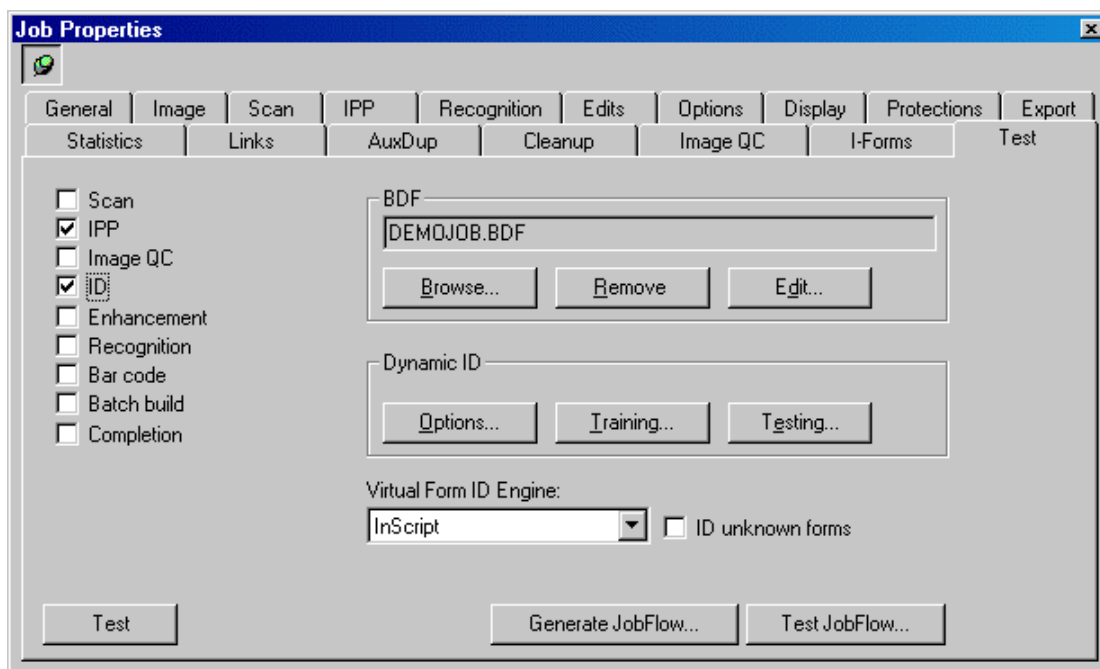
3. Select the **IPP** check box and any other processes you want to test.

4. In the **BDF** area, select the **Browse** button to search for a BDF.

5. Highlight a BDF in the list and select **Open** to choose it for testing. The file name now appears in the BDF field on the Test tab.

6. Use the **Edit** button to view the BDF to ensure that the job name on the first line is the same as the job you are currently testing. Testing IPP on a BDF that is not in your current job will apply current IPP settings to that BDF.

Caution: Running the IPP test option against a BDF permanently alters the images in that BDF. Multiple testing of the same BDF may corrupt the referenced images by re-applying IPP settings. These images could become unusable due to excess feature removal or image shifting. Therefore, we recommend that you copy all image files referenced in the BDF to another folder before you run your test.

7.   Select the **Test** button to test your IPP options.

Testing Color IPP

Follow these steps to test Color IPP at the form level in Designer.

1.   Open a job in Designer and then a form that uses color images.

2.   Display the Form Properties, Test tab.
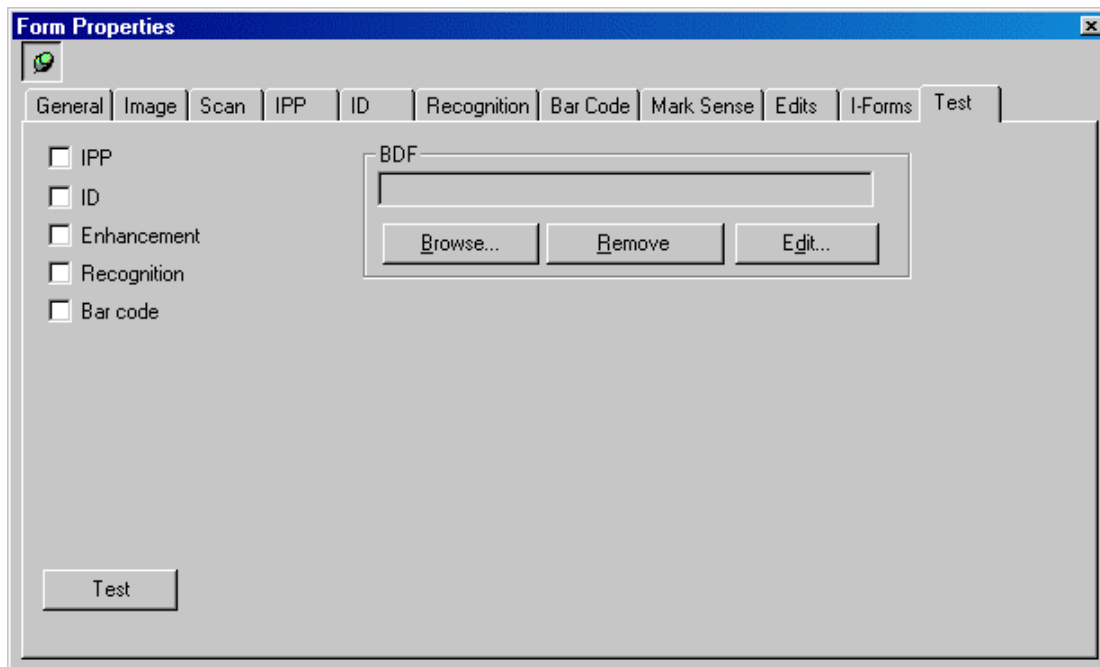
3.   Select the **IPP** check box.

4.   Select the **Browse** button to locate and select a BDF that contains color image information.

5.   Select the **Test** button to run IPP over the BDF.

When Color IPP processing is successful, both the color and bi-tonal (black and white) output images display in the Image Viewer.

Running IPP from Administration

Once you have defined IPP options for a job, you can add the Image Pre-Processing task to your JobFlow. See the Creating a JobFlow section for more information. If you want to run IPP separately from JobFlow, you can do so through the Administration module. Follow these steps:

1.   Open the Administration module.

2.   From the File Browser, highlight the Files folder.

3.   Select the **Show All Files** button on the toolbar. The BDF files in the Files folder display in the Contents area.

4.   Select a BDF file from the Contents area. To select more than one BDF file, hold down your [Shift] or [Ctrl] key and use your mouse to choose the files.

5.   From the Administration menu bar, select **Process > Image Processing** option. The Image Processing dialog displays.
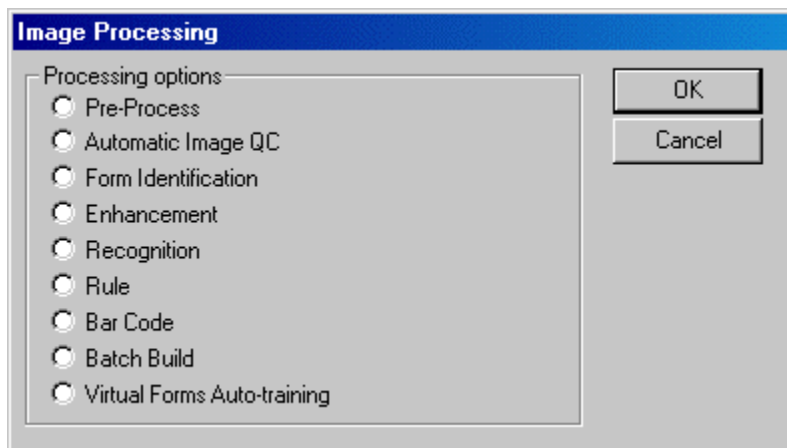
6. Select the **Pre-Process** option.

7. Select **OK**. IPP runs on the BDF file(s) you selected.

Defining Image Quality Check Properties

FormWare enables you to check and verify the quality of images in the system. The Image Quality Check (IQC) process identifies image problems such as a dark image, light contrast, holes, torn pages, or bent corners. After a problem is identified, the image can be routed to a queue for rescanning or other correction.

There are two types of IQC:

- **Automatic IQC** is run on the images after IPP and before Form Identification. Images are surveyed for problems and are rejected based on reject thresholds specified for the job in Designer. If at least one image in a BDF is rejected, Automatic IQC returns a routing code, which can then be used by JobFlow to route the BDF to a Manual IQC station. Each defective image is tagged in the BDF with a reject reason for identification later. A log file is also generated, which can be used to help locate the paper documents of the defective images for the rescan operator.

- **Manual IQC** enables you to rescan, rotate, register, or invert images to correct them. You can also perform other corrective procedures, such as reordering the images in the BDF or assigning a data form to an image. Within Manual IQC you can also perform high-speed Manual Form ID. Manual IQC is generally run after IPP and Form Identification, but can also be run directly after Automatic IQC or after Image Enhancement.

Note: The Image Quality Check process is designed to run prior to the Completion process. If you run a BDF that has been processed by Completion through Image Quality Check, you may encounter conflicts.

Setting Automatic Image Quality Check Properties

IQC is a mode of the Completion process that allows you to verify the quality of scanned images, perform manual form identification, reassemble documents, and reject images. Before you can run IQC, you must create thumbnail files of the images. Thumbnail files are separate image files that display during the IQC process and are created during Image Pre-Processing (IPP).

> **Note**: To create thumbnails, select the **Generate Thumbnail** check box on the Job Properties, IPP tab. To set up Manual IQC to stop on images that were rejected during Automatic IQC, select the **Only process rejects** check box on the same tab. For more information about these IPP settings, see the Setting Job-Level IPP Properties section.

To set Image Quality Check properties:

1. Open a job in Designer.

2. Display the Job Properties, Image QC tab.

3. Select from the following check box options to enable these functions during the Automatic IQC process:

    o **Skew** - Detects whether the image is considerably skewed (i.e., not straight on the page) based on the Skew settings on the Image QC Properties dialog.

    o **Contrast** - Detects whether the difference between the black and white areas on the image is outside of a particular range (i.e., too light or too dark) based on the Contrast settings on the Image QC Properties dialog.

    o **Black Hole** - Detects whether large black areas exist on the image that are outside of Black Hole settings on the Image QC Properties dialog.

    o **Black Bent Corner** - Detects whether black bent corners exist on the image that are outside of the Black Bent Corner settings on the Image QC Properties dialog.

    o **Black Tear** - Detects whether black tears exist on the image that are outside of the Black Tear settings on the Image QC Properties dialog.

    o **Speckles** - Detects whether large speckles exist on the image that are outside of the Speckles settings on the Image QC Properties dialog.

    o **Borders** - Detects whether image borders exist that are outside of the Border settings on the Image QC Properties dialog.

    o **Only process rejects** - Displays the first image on a page with rejects to the operator when Automatic IQC is run.

4. In the **Image Thumbnail** area, set the **Height** and **Width** (in pixels) of the image thumbnail displayed when running IQC.

5. Select the **Properties** button to set the threshold ranges for each quality defect you selected. The Automatic IQC process uses these threshold settings to determine whether to reject an image. The Image QC Properties dialog displays.

6. In the **Minimum angle** field in the **Skew** area, select the minimum acceptable angle of skew of an image in degrees. If the image is oriented by an angle greater than this value, in either positive (clockwise) or negative (counter-clockwise) direction, Automatic IQC will reject it. The default value is 2 degrees. The legal range for the minimum angle is ± 45.0 degrees. You can specify the minimum angle in 0.25 degree increments.

7. In the **Contrast** area, set properties to control whether to reject an image as too light or too dark. Automatic IQC uses two methods to measure contrast: threshold method and character height. You can use both methods simultaneously. The threshold (default) method counts the number of black pixels in the image and is the most commonly used method for determining contrast.

    The character height method determines whether at least one splotch on the image is at least the size of one character. This method, used only for rejection based on lightness of the image, does not affect the *too dark* condition. To reject based on the too dark setting, you must use the **Black Threshold** option. Control the character height method by setting an

expected minimum and maximum height of a character. By default, this method is disabled. If a character is detected, the image is accepted immediately. This method is useful if you want images that contain only a few characters on a completely white background to be accepted, whereas the threshold method will reject them as *too light*. The character height method is always applied before the threshold method.

Use the following fields to set the Contrast properties:

- o **Minimum Height** - Specifies the minimum acceptable height of any splotch on the image. The range is 0 – 9,999 pixels.
- o **Maximum Height** - Specifies the maximum acceptable height of any splotch on the image. The range is 0 – 9,999 pixels. If the height of any splotch on the image is between the **Minimum** and **Maximum Height** values, the image is accepted.
- o **Black Threshold** - Specifies the maximum number of black pixels acceptable on the image. If the number of black pixels in the image exceeds the value in this field, it will be rejected as too dark. The range is 0 – 999,999,999 pixels.
- o **White Threshold** - Specifies the minimum number of white pixels acceptable on the image. If the number of black pixels on the image is less than the value in this field, it is rejected as too light. If the number of black pixels in the image exceeds the **White Threshold** value, but is less than the **Black Threshold** value, it is accepted. The range is 0 – 999,999 pixels.

8.   In the **Black Hole** area, use the following fields to locate large or fairly large black objects on the image that are roughly the shape of holes:

- o **Minimum Height** - Specifies the minimum acceptable height of a black hole on the image. The range for each of these fields is 0 – 9,999 pixels.
- o **Maximum Height** - Specifies the maximum acceptable height of a black hole on the image
- o **Minimum Width** - Specifies the minimum acceptable width of a black hole on the image.
- o **Maximum Width** - Specifies the maximum acceptable width of a black hole on the image.

9.   In the **Black Bent Corner** area, use the following fields to detect bent corners typically produced by a fold in the paper during scanning:

- o **Minimum Height** - Specifies the minimum acceptable height of a black bent corner on the image. The range for each these fields is 0 – 9,999 pixels.
- o **Maximum Height** - Specifies the maximum acceptable height of a black bent corner on the image.
- o **Minimum Width** - Specifies the minimum acceptable width of a black bent corner on the image.
- o **Maximum Width** - Specifies the maximum acceptable width of a black bent corner on the image.
- o **Closeness to Corner** - Specifies the distance of the bent black corner from the corner of the image.

A black bent corner defect must satisfy the following requirements or the image is rejected:

- o The height of the bent corner is between **Minimum** and **Maximum Height** settings.
- o The width of the bent corner on the image is between **Minimum** and **Maximum Width** settings.

- o The bent corner is located a distance less than or equal to the value in the **Closeness to Corner** field indicating the number of pixels from the corner of the image.

10. In the **Black Tear** area, set the values in the following fields to find torn black areas that start near the edge of the image:
   - o **Minimum Length** - Specifies the minimum acceptable length of a black tear on the image. The range for each of these fields is 0 – 9,999 pixels.
   - o **Maximum Length** - Specifies the maximum acceptable length of a black tear on the image.
   - o **Minimum Thickness** - Defines the minimum acceptable thickness of a black tear on the image.
   - o **Maximum Thickness** - Defines the maximum acceptable thickness of a black tear on the image.
   - o **Closeness to Edge** - Defines the distance of the black tear from the edge of the image.

   A black tear defect must satisfy the following requirements or the image is rejected:
   - o The length of the black tear in the image is between **Minimum Length** and **Maximum Length** settings.
   - o The thickness of the black tear is between **Minimum Length** and **Maximum Length** settings.
   - o The black tear is located a distance less than or equal to the value in the **Closeness to Edge** field indicating the number of pixels from the outer edge of the image.

11. In the **Speckles** area, use the values in the following fields to determine whether an image contains speckle noise. Speckles are isolated black specks on an image that create noise and degrade recognition results. There are two methods for determining speckles: the diameter method and the density method. The diameter method searches for speckles on the image of less than a given diameter and if there are enough of them, the image is rejected. The density method checks for poorly scanned or broken characters grouped closely, such as very lightly scanned and broken-up dot matrix characters.
   - o **Maximum Diameter** - Specifies the maximum acceptable diameter of a speckle on the image. The range is 0 – 9,999 pixels.
   - o **Count Threshold** - Specifies the maximum acceptable number of speckles detected on the image. The range is 0 – 999,999 pixels.
   - o **Density Threshold** - Specifies the maximum acceptable pixel density of a speckle object, measured as a percentage. A density threshold close to 100% corresponds to something, such as a broken character rather than a random series of dots. A density of close to zero percent means that dots are scattered apart.
   - o **Closeness** - Specifies the maximum acceptable distance of how close groups of speckles are near each other. The range is 0 – 9,999 pixels.

   An image is rejected if **either** of the following two conditions is met:
   - o The diameter of a speckle on the image is less than or equal to the **Maximum Diameter** value, and there are at least as many speckles on the image as indicated in the **Count Threshold** field.
   - o Groups of speckles are closer together by less than the value indicated in the **Closeness** field, and the pixel density of this grouped speckle object is greater than the setting in the **Density Threshold** field.

> **Note**: In IQC processing, the diameter method is applied first. If the image is acceptable, the density method is then applied.

12. In the **Borders** area, set the following fields to determine if a black border exists on the image:

   o **Minimum Top** - Specifies the minimum acceptable width of a border on the top of the image. The range for each of these fields is 0 – 9,999 pixels.

   o **Minimum Bottom** - Specifies the minimum acceptable width of a border on the bottom of the image.

   o **Minimum Left** - Specifies the minimum acceptable width of a border on the left of the image.

   o **Minimum Right** - Specifies the minimum acceptable width of a border on the right of the image.

   An image is rejected if any one of the following conditions is met:

   o The top border of the image is wider than the **Minimum Top** value.

   o The bottom border of the image is wider than the **Minimum Bottom** value.

   o The left border of the image is wider than **Minimum Left** value.

   o The right border of the image is wider than **Minimum Right** value.

13. Select the **Use Defaults** button to display the IQC Defaults dialog. Select either the **200 DPI** or **300 DPI** setting for the job. Images processed with the job by Automatic IQC must all be the DPI you select. 200 DPI is the default. Then, select **OK**.

> **Caution**: Use caution when setting this option because good images with the incorrect DPI can be incorrectly rejected, and bad images with the correct DPI can be incorrectly accepted.

Setting Manual Image Quality Check Properties

Manual Image Quality Check (IQC) enables an operator to view thumbnails of the images assigned to Manual IQC and verify their quality before sending them for further processing. The operator can determine whether any of the images need to be rescanned and can either rescan the images at that time or reject the images for rescanning later. Additional functions, such as rotating, registering, or inverting the images, can be applied at this time.

The image(s) in the current BDF display as thumbnails in the Manual IQC window.

By default, the thumbnail images created for Manual IQC are placed in a directory that has the same name as the directory containing the images, with a .thm extension (e.g., \FormWare\Images.thm). If you want to specify an alternate directory, you must set the THMPATH=<pathname> parameter in the BDF indicating the path to an existing directory. You can use the Batch File Initialization Data option in the Scan module to write this parameter in the BDF during a scanning or importing process.

The Manual IQC process is generally run after the IPP and Form Identification processes. However, it can work with images rejected from Automatic IQC, and therefore, can run directly after the Automatic IQC process. Although less common, Manual IQC can run after Image Enhancement as well.

Creating an Image Reject Menu

In many forms processing applications, some of the images in a given batch cannot be processed. This can be because the image is upside down, the image is not the right form type, the back of the form was scanned, etc. You can also create a list of reasons that display instead of the default reasons in the Select Reject dialog. The list of reasons is contained in a reject menu file (REJ) in the \FormWare\Jobs folder. This file has the same name as the job with a .rej extension.

Note: Because the reject reason logic is implemented within the base client, it pertains to all base client-derived components (not including Completion). Reject reason logic is applied at the record level, therefore no form-level processing occurs and the OnFormStart/OnFormEnd events are not called. However, image-level processing (and the OnImageStart/OnImageEnd events) does occur.

The items you define in the REJ file display in the Reject Selection dialog for the job during a Completion session when an operator selects the **Record > Reject** menu item in the Completion module or presses [F3]-[R].

To view or modify the reject menu file:

1. Open the job in Designer.

2. Display the Job Properties, General tab.

3. Select the **Reject Menu** button. The reject selection text file displays in the text editor.

4. Type the list of reject reasons you want to display in the Select Reject dialog during Completion. Each value must be on its own line followed by a carriage return. Each line of the reject file has a maximum of 68 characters. Additional characters will not display. You can include a maximum of 26 reasons. Each reject reason is assigned a letter of the alphabet, which translates to a limit of 26. A reject reason is stored in the record and can be programmatically retrieved using VBA or FPL.

    The record reject codes returned for a selection in the Reject Selection dialog are hard-coded within the Completion module. The reject code for the first selection is A, the second is B, and so on. You can incorporate these reject codes into the custom reject menu, so that the operator is prompted to key the letter. For example:

    A - Image Not Readable

    B - Wrong Image

5. When you are finished, select the **Save** button on the text editor toolbar to save your changes.

6. Select **Exit** from the File menu to return to Designer.

7. Select **Save All** on the Designer window toolbar to save your changes.

Understanding Reject Return Code Values

When Automatic IQC rejects an image, the image is marked in the BDF and assigned a routing code and a reject code, and a log file is produced. The BDF contains an FW_IQC= parameter on a line above the path name of the rejected image. This parameter is also written to the log file and contains a reject message and code. The following sections explain each of the codes and the log file.

Understanding IQC Reject Codes

When an image fails Automatic IQC, the BDF is assigned a reject code. This reject code can be used at a later time to determine how to address the image problem. Reject codes include:

Reject Code	Description
0	Image OK
1	Image too light
2	Image too dark
3	Skew
4	Speckles
5	Borders
6	Halftone *
7	Bent Corners
8	Orientation *
9	Hole
10	Double Feed *
11	Torn Page
12	Image stretched *
13	Image squeezed *
14	Uneven darkness *
98	Could not open/read image
99	Internal error processing image

* Not supported in this release.

Understanding IQC Routing Codes

When images are processed through Automatic IQC, the BDF is assigned a routing code. This routing code can be used to forward the BDF to the next process, such as to a scanning station if an image needs to be rescanned. Automatic IQC routing codes include:

Routing Code	Description
0	Image OK
1	Page requires rescan
2	Page corrected—an enhanced image is created and the image is rerouted to the first post-SCAN JobFlow task
3	Image should be rerouted to IPP
4	Image should be re-routed to manual verification, with possible rescan

Understanding IQC Entries in the Log File

If at least one image in the batch fails Automatic IQC, the BDF is modified and the BDF name log file is appended to or created in the \FormWare\Files directory. The log file entry contains sufficient information for an administrator to identify and locate the paper document associated with the defective image prior to the Rescan task.

Batches with rejected images are routed to Manual IQC where an operator can select the **View > File Log** menu option to view the contents of the log file. Once a log file has been viewed and closed in Manual IQC mode, the log file is deleted.

The following is the syntax of the log file entry:

```
FW_IQC={
BCHNAME=\PATH\<filename>.BDF
   IMAGENAME=\PATH\IMAGE.EXT
   SEQUENCE=#
   FW_IQC=message,code
}
```

Any other information that would aid in locating the paper document for rescan is also included in the log file. The log file contains the following entries:

- BCHNAME - The full pathname of the BDF.

- IMAGENAME - The full pathname of the image file.

- SEQUENCE - The count of the number of images preceding the number of images in the current batch.

- FW_IQC - The reason for rejecting the image.

Following is a sample log file:

```
FW_IQC={
   BCHNAME=C:\FORMWARE\FILES\IQCTEST.BDF
   IMAGENAME=Z:\QA\TEST\IQC\IQC300\AMBC2050.TIF
   SEQUENCE=5
FW_IQC=Speckles,5
}
FW_IQC={
   BCHNAME=C:\FORMWARE\FILES\IQCTEST.BDF
   IMAGENAME=Z:\QA\TEST\IQC\IQC300\CTBC2100.TIF
   SEQUENCE=1
   FW_IQC=Skew,6
}
```

Testing Image Quality Check

Once you have defined IQC, you can test your Automatic and Manual IQC settings. To do this:

1. Open a job in Designer that uses IQC.

2. Display the Job Properties, Image QC tab for the job.

3. Select at least one check box on the Image QC tab to run Automatic IQC.

4. Display the Job Properties, Test tab.

5. Select the **Image QC** check box to run Manual IQC.

6. In the **BDF** field, select the **Browse** button to find the BDF you want to test against.

7. Select the **Test** button. The thumbnails for the images in the BDF display.

8. Check the images and close the Completion window.

For more information on the toolbar options, see the Running Manual IQC section.

Setting Form Identification Properties

Form Identification (Form ID) is the FormWare component that relates production images with a data form. If you are processing batches containing images of more than one type of form, you should include the Form ID process within your application. The Form ID component uses different methods to identify the paper form and load the Form Definition File (FDF) for recognition processing or key from image. Without Form ID, document preparation would be more labor-intensive, increasing processing time and costs.

You can ID a form using a physical attribute on the image, such as a bar code or a document identifier number, shown in the examples below. You can also identify a form based on the file size of the image or its height and width in pixels.

Understanding the Form ID Methods
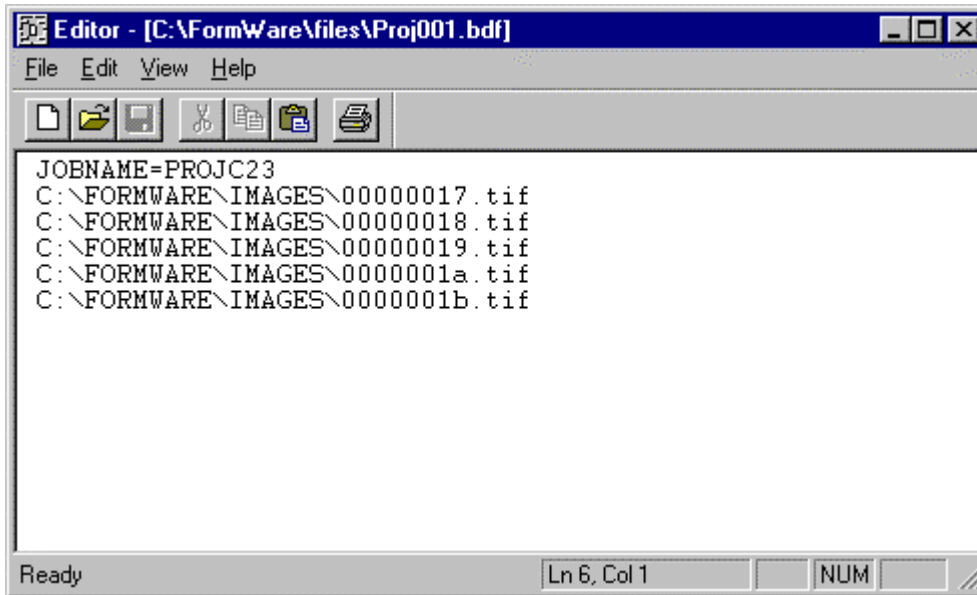
Within FormWare, you can use two types of Form ID: Automatic and Manual.

Automatic Form ID modifies the BDF to include the data form number with each image and is performed prior to a Completion session. There are four types of Automatic Form ID:

- **Dynamic Form ID**, which trains a set of assigned form images to identify the production images.
- **Feature Form ID**, which uses elements of the setup image specified in the Designer module.
- **Implicit Form ID**, which applies either Feature or Dynamic Form ID to the first image in a document. The remaining images are assigned form identification numbers sequentially without applying identification technology.
- **Virtual Form ID**, which extends the basic Form ID hierarchy of job definition and form definition to variant definition. Form ID uses Virtual Classification by making use of multiple freely located fields of textual data in unique positions on a layout variant of a form type (e.g., invoice or claim) to determine sub-classes and variants.

Manual Form ID requires the Completion operator to determine which image corresponds to which data form within the forms processing application. This manual process is used when the Automatic Form ID processes have failed to identify the image. Manual identification is performed during either the Image QC/Rescan mode or the Identify Images mode of Completion.

The numbers for the forms are placed before the image file names in the BDF. Here is an example of a BDF file prior to Form ID:
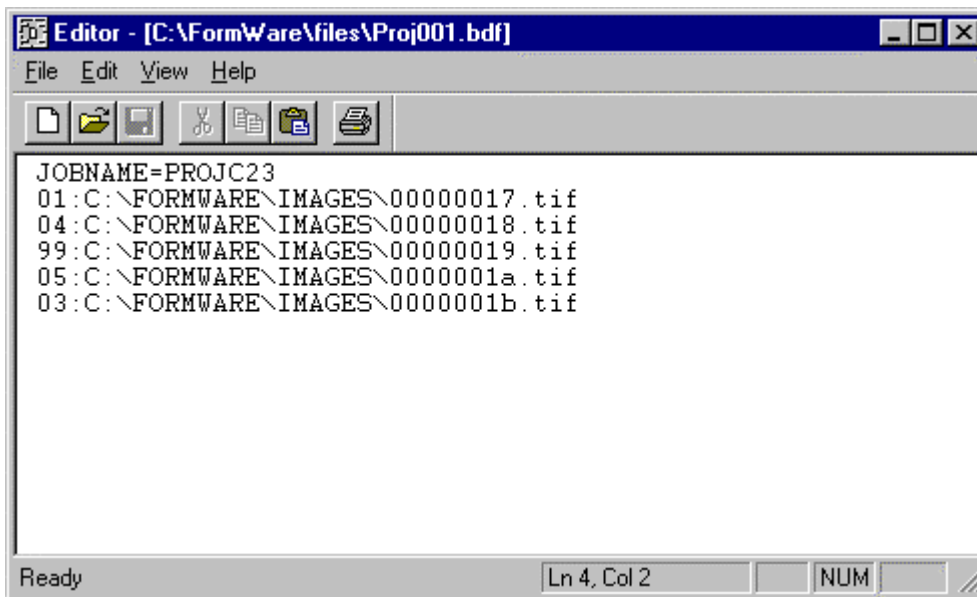
```
Editor - [C:\FormWare\files\Proj001.bdf]                    _ □ ×
File  Edit  View  Help

 ┌─┐┌─┐┌─┐  ┌─┐┌─┐┌─┐  ┌─┐
 │ ││ ││ │  │✂││ ││ │  │ │
 └─┘└─┘└─┘  └─┘└─┘└─┘  └─┘

JOBNAME=PROJC23
C:\FORMWARE\IMAGES\00000017.tif
C:\FORMWARE\IMAGES\00000018.tif
C:\FORMWARE\IMAGES\00000019.tif
C:\FORMWARE\IMAGES\0000001a.tif
C:\FORMWARE\IMAGES\0000001b.tif




Ready                                    Ln 6, Col 1      │  NUM  │
```

Here is the same BDF file after Form ID:

```
Editor - [C:\FormWare\files\Proj001.bdf]                    _ □ ×
File  Edit  View  Help

 ┌─┐┌─┐┌─┐  ┌─┐┌─┐┌─┐  ┌─┐
 │ ││ ││ │  │✂││ ││ │  │ │
 └─┘└─┘└─┘  └─┘└─┘└─┘  └─┘

JOBNAME=PROJC23
01:C:\FORMWARE\IMAGES\00000017.tif
04:C:\FORMWARE\IMAGES\00000018.tif
99:C:\FORMWARE\IMAGES\00000019.tif
05:C:\FORMWARE\IMAGES\0000001a.tif
03:C:\FORMWARE\IMAGES\0000001b.tif




Ready                                    Ln 4, Col 2      │  NUM  │
```

Form number 99 represents a failure to identify the image with a data form unless you have configured your system to use more than 100 form types. See the Expanding the Form Type Limitation section for more information on how to configure your system to use more than 100 form types.

Expanding the Form Type Limitation

The default system configuration for all FormWare systems limits the number of form types you can have within a job to 100. You can configure your system to expand this limitation to 16,000 forms per job.

When you expand the form type limitation to 16,000 forms, you will note some minor changes to the Form ID process. For example, unidentifiable forms now have a 00000 designation instead of

a 99 designation. This change will be reflected in the BDF file once the Form ID process is complete. A reference to an image file in the BDF that previously looked like:

```
01: C:\FORMWARE\IMAGES\IMG0001.TIF
```

now looks like:

```
00001: C:\FORMWARE\IMAGES\IMG0001.TIF
```

To reconfigure your system, you must add a parameter to either the System.ini file (to change the form type limitation for the entire FormWare system) or to the job INI file (to change the limitation for a job).

Follow these steps to expand the form type limitation:

1. From the Administration window, display the contents of the INI folder in the File Browser.

2. Select the appropriate sub folder, either the \INI\System folder or the \INI\Job folder to edit the appropriate INI file.

3. Double-click on the INI file you want to edit. The INI file displays in a text editor.

4. Add the following parameter followed by a carriage return to the INI file:

```
FEX=Y
```

5. Save your changes and exit the text editor.

Assigning a Specific Form Number to Unknown Forms

Form ID uses an INI setting FORCE_UNKNOWN=<form number> to assign a specific form number to all failed images during Form ID. The FORCE_UNKNOWN parameter is implemented by a call to the base components GetProperty() function. You can set this parameter in the System.ini file, job INI file or run it as a command line argument that calls the Form ID component. If your FormWare application contains only one job, set this parameter in the System.ini file. If your FormWare application contains more than one job, set this parameter in the job INI file.

Using Automatic Form ID

If your application processes multiple types of forms, you can configure Form ID to determine the data form type as the images are processed. This is referred to as Automatic Form ID. You do not need Automatic Form ID if you manually batch the same form type together and process only that one form type in each run.

You can configure Automatic Form ID to optimize identification accuracy and efficiency. In general, use the quickest and most reliable method to solve your identification problems, before you attempt to configure a more complex solution. To address configuration ease and processing speed, FormWare provides:

- Form-level identification based on image size or dimensions.
- Dynamic form identification based on automated training of a neural network from sample images.
- Feature form identification based on specific image features identified by the developer.
- A combination of Dynamic and Feature form identification, where Dynamic Form ID narrows down the possible form types, and then Feature Form ID is used to correlate forms with similar appearance.

Setting Dynamic Form ID Properties

Dynamic identification identifies the form based on training from a specified set of form images. Dynamic Form ID examines each image in the batch, identifies the FDF, and determines the orientation of each image. This process uses neural networking to examine the entire image to recognize form types during the identification process.

Dynamic Form ID uses a Training Parameters file and an Image List file to train the neural network and create an FDF. Training is the process by which a series of form images are analyzed and the neural network is optimized for identifying forms. You use Designer to assign the training form images and train them.

The following file types are used in the Dynamic Form ID process:

- **Training Parameters file (TRN)** - The TRN is an ASCII file that contains the names of the different data forms, the identification confidence level required to accept that image during production, and other parameters that can tune and optimize Form ID performance. It contains confidence and spread thresholds, information about how each form type was scanned, and a data form name that matches the data form name you specified in the Designer module, which identifies the FDF. The TRN file has the same name as the job that generates it with a .trn extension. See the Editing the Training Parameters File section for more information.

- **Image List file (.inp)** – This file lists the paths, filenames, and data form names of each form type image in the training set, along with its correct form identification. This file has a .inp extension.

- **Form Identification file (FDF)** - The FDF is the result of the training process. It drives the Form ID process during production. This file has a .id extension.

After the FDF is created, you can test it on the current image or on an input file that references different images than those used during the training process. After testing Form ID on an image or image list, a log file is generated containing statistics that help you achieve the acceptance and accuracy levels you need. Repeat the testing process until you attain the desired levels you want.

To configure Dynamic Form ID, you must select training images and then train those images over your forms. When your the application contains only two different form types that are easy to distinguish, you can quickly train five or less images of each form type (10-20 is average). The key to training images is that they represent the variety of the form types expected in production, especially if dropout ink is used on the form. Use images with all the different variations or patterns of data found in production.

Selecting Images for Training

Before you select training images, make sure that you have a number of test images available (in the \FormWare\Images folder) then follow these steps:

1. Open a job in Designer for which you want to define Dynamic Form ID.

2. Open a form within the job and display the Form Properties, ID tab.

3. Select the **Training Images** button. The Add or Remove Training Images dialog displays.

4. Select the **Add** button to display the Open dialog and select the training images. The training images you specify must be of the same type as the data form. Training with 10-20 images per image type is average.

5. If you specify a wrong image, highlight the image in the list and select the **Remove** button.

6. After you add all your training images, select **OK** to return to the data form.

Move to another data form and repeat the previous steps to add the training images for that data form. Continue adding training images until you have done so for all data forms in your job.

Training Images

Training images allows you to have Form ID determine the unique distinguishing characteristics of each form. Once you have added a set of training images, you are ready to train them for Dynamic Form ID. Training images on your data form creates a TRN file.

Follow these steps to train images for Dynamic Form ID:

1. Open a job within the Designer module and display the Job Properties, Test tab.

2. Select the **Training** button in the Dynamic ID area to start the training process. The Dynamic Form ID - Training dialog displays.

3. Select from the following check boxes in the **Display** area to select the items displayed during processing.
    o **Each Image** - Displays each TIFF image during the training process.
    o **Each Image's Results** - Displays detailed results from the training process for each image in the Each Image's Results area once training is complete.
    o **Summary of All Results** - Saves a summary of results for all images processed and displays this information in the Summary of All Results area once training is complete.

4. Select one of the following **Speed** options to control at what speed the images are processed:
    o **Manual** – Stops processing between each image until you select the **Continue** button.
    o **Timed** – Pauses briefly between processing each image.
    o **Full Speed** – Processes all images in the BDF without pausing.

5. Select the **Start** button to begin training. If you selected the **Each Image** check box, a progress dialog displays for each image as it is processed. If you selected the **Each Image's Results** and **Summary of All Results** check boxes, information displays in those areas on the dialog when the training is completed.

    The **Each Image's Results** area displays the following information for trained images:
    o **Image** - Full path of the image file used in the training process.
    o **Processed** - Total number of images trained.
    o **ID** - Image form name as specified in the Image List file.
    o **Net Level** - Level of neural networking utilization to identify the image form type. This refers to the "quadrant" method for Dynamic Form ID. For example, Level "0" means that the "Full field" was used to process this image. Level "1" means one of the 4 image quadrants was used. Level "2" means that one of the 4 sub-quadrants within the level 1 quadrants was used.
    o **Region** - Displays the area of the image used to identify the form type. The full field region indicates that the full recognition zones were used. Because you will not use any regions other than the full field region, this result will not change.

    The **Summary of All Results** section of the Dynamic ID – Training dialog displays the following information relative to the results of all of the trained images.

- o **Log for** - Full path of the Form Identification file.
- o **ID** - Image form name as specified in the Image List file.
- o **Level** - Level of neural net utilization to identify the image form type. This refers to the "quadrant" method for Dynamic Form ID. For example, Level "0" means that the "Full field" was used to process this image. Level "1" means one of the 4 image quadrants was used. Level "2" means that one of the 4 sub-quadrants within the level 1 quadrants was used.
- o **Region** - Area of the image used to identify the form type. The full field region indicates that the full recognition zones were used. Because you will not use any regions other than the full field region, this result will not change.
- o **Image** - Full path of the image file used for training.

6. Select **OK** to exit the Training dialog and return to the Designer module.

Testing Dynamic Form ID

Testing the set of training images allows you to test Form ID and review the results of individual images and overall statistics. After you have selected and trained the images for your data form, you are ready to test the identification against images in a BDF.

Caution: Do not use the **Test** button on the Form Properties, ID tab to test Dynamic Form ID or you may receive an error. The Form Properties Test button is only for Feature Form ID.

To test images for Dynamic Form ID within a BDF:

1. Open a job in Designer containing a form that you have trained images for Dynamic Form ID.

2. Display the Job Properties, Test tab.

3. Select the **Browse** button in the BDF section to display the Browse Files dialog where you can select a BDF.

4. Check the **ID unknown forms** check box, if you want to force the application of Feature Form ID if the Dynamic Form ID process fails. This option can only be set at the job level. See the Running Feature Form ID After Dynamic Form ID Fails section for more information on this setting.

5. Select the **Testing** button in the **Dynamic ID** area. The Dynamic ID – Testing dialog displays.

6. Set the desired parameters in the **Display** and **Speed** areas (as you did for training) for the testing process.

7. Select the **Start** button to begin testing the images in the BDF. If you selected the **Each Image** option, a progress dialog displays for each image as it is processed. If you selected the **Each Image's Results** and **Summary of All Results** options, information displays in those sections of the dialog when the training is complete.

   The **Each Image's Results** area displays the following information for tested images:
   - o **Image** - Full path of the image file used in the testing process.
   - o **# Processed** - Total number of images tested.
   - o **ID** - Image form type as specified in the input file.
   - o **Conf** - Confidence of identifying this image on a scale of 0 to 100 percent.

- o **Spread** - Difference between the confidence of the first identification choice and the second choice.
- o **Orientation** - Number of degrees the image was rotated to perform Form ID.
- o **Second Choice** - Name of the image that is the second identification choice and its confidence level.

The **Summary of All Results** section displays information relative to the results of all of the images tested.

- o **Total Images** - Total number of images tested.
- o **Accepted** - Number and percentage of the tested images that were accepted.
- o **Accepted and Correct** - Number and percentage of the tested images that were accepted and correct as recognized.
- o **Rejected** - Number and percentage of the tested images that were rejected.
- o **Rejected but Correct** - Number and percentage of the tested images that were rejected even though they were correctly recognized.

8. Select the **Review** button to display the Dynamic ID – Results dialog. You can analyze these results to determine whether you need to fine tune any of the training parameters and then retest to obtain the Form ID results you want. You can save the Image Summary as another view or as an Image List Input (INP) file for additional testing and training. The Dynamic ID - Results dialog displays the following information.

- o **Statistical Summary** – Displays a summary of the current results. Summary information is regenerated each time you use the Apply Filters and Show Only Differences options.
- o **Total Images** - Total number of images tested.
- o **Accepted** - Number and percentage of the tested images that were accepted by the Dynamic ID training process based on the form you created using the setup image.
- o **Accepted and Correct** - Number and percentage of the tested images that were accepted and correct as recognized by the Dynamic ID training process based on the form you created using the setup image.
- o **Rejected** - Number and percentage of the tested images that were rejected by the Dynamic ID training process based on the form you created using the setup image.
- o **Rejected but Correct** - Number and percentage of the tested images that were rejected even though they were recognized correctly.
- o **Image Summary** - Displays the image-by-image test results from the Dynamic Form ID test you just ran. If you applied any Column Filters or a Difference File on the Dynamic ID - Results dialog, this view may contain only a subset of the items from the test results file. To view the .TIF image for a row of statistical data, you can double-click anywhere on that row. A separate image display window is used to show the image. To close the image display window, activate the display window and press [ESC].
- o **ID-ed as (Identified As)** - Form type results generated by the recognition process.
- o **Proc Type** - Processing type used to identify the image. Possible values of the processing type are D - Dynamic Form ID processing only was run on the image, F - Feature Form ID processing only was run on the image, and B – Both Dynamic *and* Feature Form ID processing were run on the image.
- o **A/R** - Letter that indicates whether the form was accepted (A) or rejected (R).
- o **Conf** - Percentage of assurance in recognition accuracy of the Form ID process.

- o **Spread** - Difference between the confidence of the first identification choice and the second choice.
- o **Conf Thresh** - Minimum confidence allowed (displayed as a percentage) before the form is rejected.
- o **Spread Thresh** - Minimum amount of spread allowed (displayed as a percentage) before the form is rejected.
- o **BT** - Minimum number of black pixels the entire image must have to be classified as non-dropout. A form classified as non-drop out does not have any drop out ink.
- o **Black Cnt** - Actual number of black pixels on the image.
- o **Scan Class** - Numeric value that represents the scan class for the image. Form ID reserves scan classes 0-7 internally. The scan class defines which category the image has been placed into by form ID based on the image dimensions, DPI, and whether it is dropout or non-dropout. For example, an image that is 1500 pixels x 2500 pixels, at 300 DPI, and is dropout might be automatically assigned to scan class "8". Another image that is 1500 x 2500, 300 DPI, and non-dropout would be assigned to scan class "9." The scan class assignment is initially done at training time.
- o **Drop Type** - Indicates whether the form is in dropout ink or non-dropout ink. An "N" indicates a non-dropout form, a "D" indicates a dropout form.
- o **Orient** - Numeric value (in degrees) that represents how much the image was rotated after scanning to obtain its current orientation within the image viewport.

Note: When processing color images, Dynamic Form ID automatically corrects and overwrites the color images that have incorrect orientation using the same orientation correction applied to the bi-tonal images. When Dynamic ID detects that the color image is oriented at 90,180, or 270 degrees, it calls the FWColor.DLL, which overwrites the incorrectly oriented color image. Additionally, certain parameters need to be set in the Training Parameters file (TRN). See Editing the TRN File for Color Image Processing section for further information.

- o **Quadrant** - Region of the form used for form identification.
- o **First** - Highest confidence form type choice resulting from Form ID.
- o **Second** - Second highest confidence form type choice resulting from Form ID.
- o **Others** - Any choices in addition to the first and second choices.
- o **Image** - Full path to the location of the image file.

9. To specify a column filter, select the desired data cell below the filter column title and type a filter expression in the **Column Filters** field. Column filters allow you to filter out items from the Image Summary view so you can view specific groups of entries contained in your results. Then, select the **Apply Filters** button.

Note: You must type a space between all operators and data. For example, in the filter expression, "!= formware," there is a space between "!=" and "formware."

This allows you to update the data in the Image Summary based on the Column Filters and a Difference File (if you specified one). The two parts of a Filter Expression are filter function and filter value.

Use the following filter functions:

- o **=** Removes from the view any data in the column that matches the following filter value. For example, "= 47" removes from the view any data in the column that matches 47.

- o **!=** Removes from the view any data in the column that does not match the following filter value. For example, "!= HCFA" removes from the view any data in the column that does not match "HCFA."
- o **<** Removes from the view any data in the column that is less than the following filter value. For example, "< 98.0" removes from the view any data in the column that is less than 98.0.
- o **>** Removes from the view any data in the column that is greater than the following filter value. For example, "> FORM4" removes from the view any data in the column that is greater than "FORM4" (ASCII collating sequence).

10. Select the **Show Only Differences from** check box to disable or re-enable Difference filtering after you select a difference file. This allows you to filter out all the items that have not changed between test runs. Select the **Browse** button to browse for a Difference file.

Note: The View and Difference test results files are generated from tests using the same Image List file (INP). If the View and Difference test results files are generated from tests using different Image List files, they will not be comparable.

11. Review the results and select **OK** to exit the dialog and return to the Designer module.

Note: To view a TIFF image, double-click anywhere on the row and the image displays in a window.

When you test the images in a BDF using this method, form numbers are not inserted into the BDF file. You should only use the **Testing** button in the Dynamic ID area as a diagnostic tool. Form numbers are written to the BDF when you select the **ID** option on the Job Properties, Test tab and then select the **Test** button.

Editing the Training Parameters File

If you are performing Dynamic Form ID, you can edit the Training Parameters file. This file is in the \FormWare\Jobs folder, and its name is based on the name of the job that created it with a .trn extension.

The TRN file contains:

- For each form type to be identified, a form name that corresponds to the form name on the Form Attributes dialog box. This identifies the FDF to be used to process that form type.
- Optional confidence and spread thresholds.
- Optional information about how each form type was scanned. For example, whether the form is in dropout or non-dropout ink and the DPI at which it was scanned.
- InScript Define provides a sample training parameters file in text editor to use as a template.

There are three levels of parameters within the Training Parameters file.

- **FIDINFO level** – The lowest, most general level, contains global parameters for all types of forms and scan classes.
- **SCAN level** – The middle level is a user-created SCAN class, which overwrites the FIDINFO level parameters.
- **FORM level** – The highest level contains parameters for the form type. You can nest the FORM level within the other two levels. The FORM level always takes precedence over

the information in the FIDINFO level and can override SCAN-level parameters, if it is nested within the SCAN class.

The following training parameter file skeleton shows the complete hierarchy supported:

```
FIDINFO = {
   .......
   .......
   SCAN = {
           .......
           .......
           FORM = {
           ......
           ......
                   }
   }
           FORM = {
           .....
           .....
                   }
   }
```

The following shows an example training file where the threshold parameters CONFTHRESH and SPREADTHRESH are set at various nesting levels:

```
FIDINFO = {
                   CONFTHRESH = 0.82F
                        SPREADTHRESH = 0.02F
                        FORMNAME = FORM_A
   SCAN = {
                        SCANCLASS = BC_CLASS
                        SPREADTHRESH = 0.04F
                        FORMNAME=FORM_B
                        FORM = {
                                FORMNAME=FORM_C
                                CONFTHRESH=0.78F
           }
           }
   FORM = {
                   SPREADTHRESH=0.06F
           }
   }
```

In the previous example, "FORM_A" will be positively identified with CONFTHRESH = 0.82F and SPREADTHRESH = 0.02F. "FORM_B" will be positively identified with CONFTHRESH = 0.82F and SPREADTHRESH = 0.04F. "FORM_C" will be positively identified with CONFTHRESH = 0.78F and SPREADTHRESH = 0.02F. Form "D" will be positively identified with SPREADTHRESH = 0.06F.

Note: In the example, the scan class has been named via the command line SCANCLASS = BC_CLASS. The SCANCLASS keyword is used as a tag for "user commenting" purposes only and can be left out of the training file. It is not currently used to define classification except in backward compatibility with older training files.

To edit a TRN file:

1. Open a job in Designer containing a form that you have trained and tested images for Dynamic Form ID.

2. Display Job Properties, Test tab.

3. Select the **Options** button in the **Dynamic ID** area. The TRN file for this job displays in the editor window.

The FIDINFO block is the main block of the file. The parameters in this block apply globally to all form types and scan classes. Three main global parameters are specified in this block. These parameters are defined as follows:

- o **CONFTHRESH** - Specifies the confidence threshold between 0.000F and 1.000F inclusive. The "F" in the CONFTHRESH value stands for float. You can change this parameter in the file. If the confidence level for the form is lower than this value, the form is rejected.
- o **SPREADTHRESH** - Specifies the difference in confidence between the first and second guesses for Form ID. You can change this parameter in the file.
- o **FORMNAME** - Specifies the name of the FDF.

4. Make any necessary changes to the file.

5. Save and exit the file.

6. Select the **Save All** button on the Designer window toolbar.

Editing the TRN File for Color Image Processing

When you use Dynamic Form ID to process color images certain modifications must be made to the TRN file to rotate color JPG images during processing. You must train Dynamic Form ID using a TRN file that contains the DISABLE_ORIENTATION = 0 parameter. Additionally, all of your training images must be set to zero rotation so that Form ID will know when these images are rotated during processing.

Perform the following steps to configure your Dynamic Form ID training file to perform rotation correction on color JPG images:

1. Follow the instructions to access the TRN file in the Editing the Training Parameters File section. The default TRN file provided with FormWare displays as follows.

```
FIDINFO = {

        CONFTHRESH = 0.80F

        SPREADTHRESH = 0.08F

        FORMNAME = COLOR

        FORMNAME = COLOR2

}
```

2. Add DISABLE_ORIENTATION = 0 before the form name list. You must leave single spaces around the = sign. For example:

```
FIDINFO = {

        CONFTHRESH = 0.80F

        SPREADTHRESH = 0.08F

        DISABLE_ORIENTATION = 0

        FORMNAME = COLOR

        FORMNAME = COLOR2

}
```

3. For some form types, images that are rotated 180 degrees look very much like right side up images. Therefore, you may want to lower the confidence and spread thresholds used for only the particular form type that is usually rotated 180 degrees. For example:

```
FIDINFO = {
```

```
            CONFTHRESH = 0.80F

            SPREADTHRESH = 0.08F

            DISABLE_ORIENTATION = 0

            FORM = {

                    FORMNAME = COLOR

                    CONFTHRESH = 0.65F

                    SPREADTHRESH = 0.05F

            }

            FORMNAME = COLOR2

    }
```

4. Save and exit the TRN file.

5. Select **Save All** on the Designer window toolbar to save your changes.

Using VBA with Dynamic Form ID

During Dynamic Form ID, the OnFormStart event fires twice. Form ID fires the OnFormStart event initially on every unknown form and then once the Form ID is determined, the OnFormStart event becomes true again. For example, if you have VBA code that executes during Form ID (e.g., to display the form number and record number) and the form number is 0 on record 1 the first time the OnFormStart event fires, then after dynamically determining the correct form number, the OnFormStart event will fire again on record 1 and display the correct form number.

Setting Feature Form ID Properties

Feature Form ID allows you to compare elements of the setup image to the production images to determine form type. You can perform Feature Form ID based on image size, image dimension, or specific features that display on the image. You also use Feature Form ID to identify those forms that fail Dynamic Form ID and have similar features.

When you use the Field option of Feature Form ID, the Registration and Removals functions are internally applied to the image as necessary to locate the data for the assigned Feature Form ID type.

Setting Feature Form ID Based on Image Size

You can set up Feature Form ID to identify a data form with images based on the image file size. Use this method when you process images whose file sizes differ greatly when they are produced after scanning, such as a non-drop out form image versus a drop out form image.

To set up Feature Form ID based on image size:

1. Verify that a setup image, which is representative of the file size for your production images, exists in the FormWare system. If you have already scanned a setup image, select the folder in the File Browser on the Administration window where the setup image is stored. The default folder for setup images is the \FormWare\Forms folder. You can identify setup images by their .sup extension.

2. Display the contents of the FormWare\Forms folder and select the **Show All Files** button to display all the files in the folder. Then select the **Details** button and record the file size information of the setup image.

3. From Administration, select the **Designer** toolbar button to open Designer.

4. Open the existing job and form for which you want to define Feature Form ID.

5. Display the Form Properties, ID tab for the form.

6. Select the **Size** check box in the **Type** area.

7. In the **Size** field, type the byte size of the setup image you wrote down within Administration.

8. In the **Confidence** field, select an appropriate percentage of accuracy, between 0-100. A higher number indicates that the image size specified for the setup image and the production images must match more closely.

9. Select the **Test** button to test the Feature Form ID settings. A dialog displays informing you whether a positive identification was made based on the image size.

10. Display the Job Properties, Test tab to test a batch containing all the data forms in your job simultaneously. This will verify that no false positive identifications are being made using this Form ID method.

11. If the test from Designer is unsuccessful, an error message displays. If you test on the job level or in a production environment, unsuccessful Form ID returns a form number 99 for the image, which is written into the BDF file.

12. Continue to adjust the **Size** and **Confidence** values until you achieve a successful test.

Setting Feature Form ID Based on Image Dimension

You can set up Feature Form ID to identify a data form based on the expected width and height of production images. Use this method of form identification when you process images of varied dimensions.

To set up Feature Form ID based on image dimension:

1. Open a job and form in Designer that will use Feature Form ID based on image dimension.

2. Verify that a setup image, which is representative of the image dimension for the production images, is loaded for the data form in the Image Viewport.

3. Right-click on the setup image in the Image Viewport to display the Image Properties dialog. This dialog contains the width and height measurements of the setup image. Make note of these values.

4. Display the Form Properties, ID tab.

5. Select the **Dimensions** check box in the **Type** area.

6. In the **Width** and **Height** fields, type the expected image dimensions in pixels.

7. In the **Confidence** field, select an appropriate percentage of accuracy, between 0-100. A higher confidence value indicates that the dimensions for the setup image and the production images must match more closely.

8. Select the **Test** button to test Feature Form ID settings based on image dimension. A dialog displays to inform you whether the test was successful.

9. Display the Job Properties, Test tab to test a batch containing all data forms in your job simultaneously. Verify that no false positive identifications are being made using this Feature Form ID method.

10. An error message displays if the test is unsuccessful. If you test on the job level or in a production environment, unsuccessful Form ID returns a form number 99 for the image, which is written into the BDF file.

11. Continue to adjust the **Dimension** and **Confidence** values until you achieve a successful test.

Setting Feature Form ID Based on Image Attributes

You can set up Feature Form ID to identify a data form based on unique features or attributes on the image, such as text, bar codes, or the presence of data. When you identify forms based on image attributes, you can specify more than one attribute per data form. Each identifying attribute is zoned and associated with a Form ID field type.

You can further specify properties for each attribute in the field properties. Once you place a Feature Form ID field on your data form using the steps in this section, refer to the sections that follow to assign the Form ID field a particular type of image attribute. During production, each data form type is examined in the order that the data forms are defined in the job. Then, the Feature Form ID fields on each data form are executed in the order in which they appear in the form.

The image is considered to be of the specified data form type upon successful attribute identification, or when a rule associated with a Form ID field returns an Accepted status. This is true even if a later Form ID zone on a subsequent data form also positively identifies the image.

To set up Feature Form ID field based on an image attribute:

1. Open a job and form in Designer that will use Feature Form ID based on an image attribute.

2. Verify that a setup image, which is representative of the production images, is loaded for the data form in the Image Viewport area of the Designer window.

3. Display the Form Properties, ID tab for the form.

4. Select the **Field** check box in the **Type** area.

5. Return to the data form area on the Designer window.

6. Select the **Form ID** button on the Field Types toolbar and click on the data form. A green, one-character field, which represents the Form ID field, appears on the data form. This Form ID field is merely a placeholder and does not display to the operator during a Completion session.

7. With the Form ID field selected, click on the setup image in the Image Viewport and draw a zone around the attribute you want to use for Feature Form ID.

Caution: Use care as you place the Form ID zone on the image. This zone is examined on all images processed for the job. If there is any data in the zone area that you are not trying to identify, misidentification can occur. As you place the Form ID zone on the setup image, you should use different X and Y coordinates for different data form types.

8. Display the Field Properties, ID tab for the Form ID field.

9. Set additional properties for the Form ID field using the fields and options on this tab (see the sections that follow).

Defining a Recognition Image Attribute

Feature Form ID based on a recognition attribute applies a recognition engine to the area on the image zoned for the Form ID field. It then can compare the recognition results to a specified text string or run a validation rule to accept or reject the data for the field. This means that you can perform Feature Form ID based on the name of a form or some other value that can be identified using recognition technology. The following is an example of a text string that might be used for recognition Feature Form ID:

Follow these steps to set up a Form ID field that uses recognition technology to identify an image attribute:

1. Follow the steps in the Setting Feature Form ID Based on Image Attributes section then proceed to the next step.

2. On the Field Properties, ID tab, select the **Recognition** option in the **Type** area.

3. In the **Data type** field, select the type of field data that the recognition engine should expect in the zoned area on the image associated with the Form ID field feature.

4. Set up Recognition Feature Form ID using one of the following methods:

    o   In the **Match string** field, type the exact syntax for the attribute in the zoned area. Note that this string data is case sensitive. You can use a question mark (?) as a wild card character.

    o   Define a rule that returns either an ACCEPT or a REJECT for the Form ID field.

5. In the **Confidence** field, type a numeric value between 0-100 that represents how confident the recognition engine should be that it has accurately recognized the zoned data. A higher numeric value indicates that the recognition engine must be more confident of identifying the zoned data on production images.

6. Display the Field Properties, Recognition tab, which contains properties for the recognition of the field. All of these properties apply to Recognition Feature Form ID. You must select a **Recognition Type**. Only the **AEG**, **InScript**, and **InScript2** recognition engines can be used for Recognition Feature Form ID.

7. Return to the Field Properties, ID tab and select the **Test** button to test form identification on the Form ID field's zoned area. A dialog displays to inform you whether the test was successful.

8. An error message displays if the test is unsuccessful. When you run a Form ID test at the job level or in a production environment, unsuccessful Form ID returns a form number 99 for the image, which is written into the BDF file.

9. Continue to adjust the property settings until you achieve a successful test.

Defining a Bar Code Image Attribute

Feature Form ID based on bar code recognition applies bar code recognition technology to the area on the image zoned for the Form ID field, and then compares it to the bar code data. When you use Bar Code recognition, you must zone a portion of the image containing a bar code and you must know what data is returned by the bar code. Bar codes are self-contained information that is encoded in the widths of printed bars and spaces. The following example of a bar code might be used for form identification.

Follow these steps to set up a Form ID field that uses Bar Code Recognition:

1. Follow the steps in the Setting Feature Form ID Based on Image Attributes section then proceed to the next step.

2. On the Field Properties, ID tab, select the **Bar Code** option in the **Type** area.

3. In the **Match string** field, type the exact syntax of the bar code data that will be returned by the bar code recognition engine. You can use a question mark (?) as a wild card character.

4. In the **Confidence** field, type a numeric value between 0-100 that represents how confident the bar code recognition engine should be that it has accurately recognized the zoned bar code data. A higher numeric value indicates that the recognition engine must be more confident of identifying the zoned data on production images.

5. Display the Field Properties, Recognition tab for the Form ID field.

6. Set the **Recognition type** field to **Bar Code**.

7. Select **InScript2** as the engine in the **Recognition engines** field.

Note: The **Other** option in the Recognition engines field enables you to use a third-party external engine for Feature Form ID Bar Code Recognition. If you select this option, FormWare assumes that all configuration and setup of this recognition engine will be external to FormWare.

8. Select the **Advanced Properties** button to display the InScript2 Advanced Field Properties dialog. See the Setting InScript2 Field-Level Properties section for more information.

9. Return to the Field Properties, ID tab and select the **Test** button to test form identification on the Form ID field's zoned area. A dialog displays to inform you whether the test was successful.

10. An error message displays if the test is unsuccessful. When you run a Form ID test at the job level (from Job Properties) or in a production environment, unsuccessful Form ID returns a form number 99 for the image, which is written into the BDF file.

11. Continue to adjust the property settings until you achieve a successful test.

Defining a Feature Image Attribute

Feature Form ID based on an image feature matches intersecting, vertical, or horizontal lines to determine the proper data form. These intersecting lines can be in the form of a "T," "+," "L," or any combination as long as they are distinctive lines with an intersecting point. The following is an example of intersecting lines that can be used for Feature Form ID.
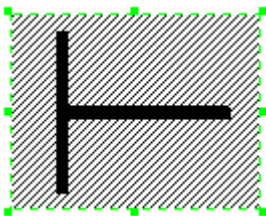
To set up a Form ID field that uses an image feature:

1. Follow the steps in the Setting Feature Form ID Based on Image Attributes section then proceed to the next step.

Note: Verify that recognition zone for this type of Form ID field contains two clear intersecting lines.

2. On the Field Properties, ID tab, select the **Feature** option in the **Type** area.

3. In the **Confidence** field, type a numeric value between 0-100 that represents how confident the recognition engine should be that it has accurately recognized the zoned data. A higher numeric value indicates that the recognition engine must be more confident of identifying the zoned data on production images.

4. Select the **Features** button to display the Registration Anchor dialog.

5. Select the button that represents the intersection type and click the matching area on the image.

6. Select the **Test Anchor** button to ensure that your anchor registers correctly and select **OK**.

Note: This process is different than assigning anchors on intersecting lines for Registration although the steps are similar. See the Registering an Image section for more information.

7. Return to the Field Properties, ID tab, select the **Test** button to test form identification on the Form ID field's zoned area. A dialog displays to inform you whether the test was successful.

8. An error message displays if the test from Designer is unsuccessful. When you test Form ID at the job level (using Job Properties) or in a production environment, unsuccessful Form ID returns a form number 99 for the image, which is written into the BDF file.

9. Continue to adjust the Form ID field properties until you achieve a successful test.

Defining a Data Presence Image Attribute

Feature Form ID based on data presence is used for a very basic type of form identification. It compares the ratio of white pixels to black pixels, regardless of shape, within the Form ID field recognition zone to determine the presence of data.

Follow these steps to set up a Form ID field based on the presence of data:

1. Follow the steps in the Setting Feature Form ID Based on Image Attributes section then proceed to the next step.

2. On the Field Properties, ID tab, select the **Data Presence** option in the **Type** area.

3. In the **Confidence** field, type a numeric value between 0-100 that represents how confident the recognition engine should be that it has accurately recognized the zoned data. A higher numeric value indicates that the recognition engine must be more confident of identifying the zoned data on production images.

4. Select the **Test** button to test form identification on the Form ID field's zoned area. A dialog displays to inform you whether the test was successful.

5. An error message displays if the test from Designer is unsuccessful. When you test Form ID at the job level (using Job Properties) or in a production environment, unsuccessful Form ID returns a form number 99 for the image, which is written into the BDF file.

6. Continue to adjust the Form ID field properties until you achieve a successful test.

Defining a Pattern Image Attribute

Feature Form ID based on a pattern uses a bitmap, such as a company logo, as the identifying feature on the image. Follow these steps to set up a Form ID field based on a data pattern:

1. Follow the steps in the Setting Feature Form ID Based on Image Attributes section then proceed to the next step.

Note: Verify that the recognition zone for the Form ID field contains a bitmap image.

2. On the Field Properties, ID tab, select the **Data Presence** option in the **Type** area.

3. In the **Confidence** field, type a numeric value between 0-100 that represents how confident the recognition engine should be that it has accurately recognized the zoned data. A higher numeric value indicates that the recognition engine must be more confident of identifying the zoned data on production images.

4. Select the **Test** button to test form identification on the Form ID field's zoned area. A dialog displays to inform you whether the test was successful.

5. An error message displays if the test is unsuccessful. If you test on the job level (Job Properties) or in a production environment, unsuccessful Form ID returns a form number 99 for the image, which is written into the BDF file.

6. Continue to adjust the Form ID field properties until you achieve a successful test.

Using Dynamic and Feature Form ID

You can design Dynamic and Feature Form ID to work together and produce a more efficient and accurate Form ID process. When you run the Form ID process from the JobFlow Client or from the Administration module, it runs both Dynamic and Feature Form ID; Dynamic Form ID runs first.

During Form ID, you can use only Feature Form ID, or only Dynamic Form ID if the form types are distinct. However, when using Dynamic ID and Feature ID together, Feature ID is not used on the images that fail Dynamic ID. Instead, if Dynamic ID is "almost" adequate, but some form types are similar, Feature ID can be used to distinguish between these similar form types once you have dynamically identified the form as one of those types. You do not have to set this up, it occurs automatically when you run the Form ID process.

Dynamic Form ID chooses the *winning* candidate form as the one with the highest confidence value. This form candidate must have a confidence greater than the CONFTHRESH parameter value set in the training parameter file for it to be a winning candidate. When the Dynamic Form ID component has a "second guess" of the form type and the confidence is within the SPREADTHRESH parameter, it deems the winning candidate as unsatisfactory and both form type candidates are placed into the cluster. The cluster is then forwarded to the Feature Form ID process. For example:

```
For a confthresh = 80.0 and spreadthresh_.0
Form type A has a confidence of 87.0
Form type B has a confidence of 81.0
Form type C has a confidence of 81.0
Form type D has a confidence of 70.0
Form type E has a confidence of 60.0
```

In this example, forms A, B, and C are all sent to Feature Form ID as possible form candidates (Cluster C). Feature Form ID then decides among these four forms what the true form should be

based on the Form ID field within the form definitions for these forms (e.g., does it match the recognition string for form A).

Note: As long as form types A, B, and C are considered valid FDF files in the job and contain at least one valid Form ID field type, Feature Form ID will process the test image against all three form types. For example, if form type C does not have a valid Form ID field defined, then only A and B are used. If none of the FDF files contain a Form ID field, then the image will fail Form ID, because the spread between the winning and second guess was not great enough.

Running Feature Form ID After Dynamic Form ID Fails

By default, when Dynamic Form ID fails (i.e., rejects an image), it does not automatically invoke the Feature Form ID process. Feature Form ID is only applied when the spread from Dynamic Form ID between the form number first guess and one or more of the alternate form number guesses is below the spread threshold (as set by the user). This enables the Dynamic Form ID process to reduce the number of images that are actually sent to Feature Form ID. Because Feature Form ID is a time-consuming process, this can dramatically reduce Form ID processing time.

However, there may be situations in which you want to force the application of Feature Form ID every time Dynamic Form ID fails to identify an image. For example, when you want to add new forms to a job, assign Feature Form ID fields to them, and then run Form ID without having to train them using Dynamic Form ID. Additionally, these "new" forms may fall into a category of forms that have always failed Dynamic Form ID. For example, they may have corrupted the Dynamic Form ID process or were "un-trainable" forms (such as forms that contained lots of variability).

To force the use of Feature Form ID after Dynamic Form ID fails:

1. Check the **ID unknown forms** checkbox on the Job Properties, Test tab. This forces the use of Feature Form ID after every failed Dynamic Form ID attempt.

Note: This option can only be set at the job level. See the Testing Dynamic Form ID section for instructions.

Caution: When Dynamic Form ID rejects an image, it is processed through all Feature Form ID procedures for all forms that have one or more Feature Form ID fields specified on a form. This means that an unknown form (Form=99) may take considerable processing time (on average) if there are a multitude of forms in the job that have Feature Form ID fields assigned to them. Therefore, use caution when using this setting or you could slow the Form ID processing time considerably.

Selecting the **ID unknown forms** checkbox also enables you to "bypass" the requirement to train every form in Dynamic Form ID. When you select this option on the Job Properties, Test tab, the Form ID component will not automatically fail on jobs that contain forms that have not been trained for Dynamic Form ID. Images for these forms will either always be rejected by Form ID (if they have not been assigned Feature ID fields) or will be sent to Feature Form ID to be identified if the ID unknown forms option has been selected.

Setting Implicit Form ID Properties

You use Implicit Form ID when you are using a forms processing application to process a document. A document is a form containing multiples pages that you want to group together.

When you process a document using Implicit Form ID, Feature or Dynamic Form ID runs on the first page of the document first. When the form type is identified, form numbers are automatically assigned to the remaining pages in the document without performing identification on them. The form numbers for the remaining pages are determined by the form links you define. You should only use Implicit Form ID when you know that the paper forms in a document are always going to be processed in a specific order.

You can also create a job with a number of data forms organized in a document as well as additional data forms not configured in the document. In this situation, Form ID is applied to the first form of the document and each of the unrelated forms. In this way, you can process a document that has an unknown number of attachments. To do this, set up a Feature or Dynamic Form ID field for the data forms that represent the first page of the document and any attachments. Link the document data forms to establish that they are part of the document. In production, Form ID examines the images until it finds the first page of the document or identifies one of the attachments. If it identifies the first page, it assigns a form number to the first image and then assigns form numbers to the next three images without performing Form ID on them. After the data form representing the end of the document is assigned a form number, Form ID continues to examine each image until it identifies the first page of the document. At this time, it can identify any of the attachment images.

Before you perform Implicit Form ID, you must set up a forms processing application for a document (i.e., multi-page form). While you can set up more than one document in a job, the following example tells you how to configure a single document using Implicit Form ID.

1. Open the job in Designer. This job should have more than one type of data form. Implicit Form ID is best used when you have a large number of forms to identify.

2. Open the data form that is the first page of the document in the Job Browser on the Designer window.

3. Set up a Feature or Dynamic Form ID field on this data form. See the Setting Dynamic Form ID Properties section or the Setting Feature Form ID Properties section for specific steps.

4. With the Form ID field selected on the data form, display the Form Properties, ID tab.

5. Select the **Auto ID Linked Forms** checkbox. This field is used to designate the first form in a document and to define the job as a document job (i.e., one with multiple data forms).

6. To assign a document name to this data form (optional) display the Form Properties, Edits tab.

    a. Select the **Start of Document** check box.

    b. In the **Document Name** field, type the name of the document.

Note: Assigning a document name to your document is optional. You can simply check the **Start of Document** option to indicate that your application uses documents and leave the **Document Name** field blank.

7. To link the forms together, display the Job Properties, Links tab. The form numbers assigned to the identified images are determined by the form number and link values you set on this tab.

8. Highlight a data form in the Form area.

9. In the **Number** field, select the number that represents the data form. You can assign any number between 00-98 to the data form. When an image matches the data form type, the form number is written in the BDF at the beginning of the image path name.

10. To link one form to another, highlight a form name in the **Form** area and then highlight the second form in the **Link to** area. The number in the Links To column updates to reflect the link.

11. Continue to link forms in order they should be processed within the job.

Note: To include additional data forms in the job that are not a part of the document, do not include the data form in the link chain that specifies the document.

---

12. To signal the end of a document, link the final data form back to any data form in the links chain. The final data form does not need to be linked back to the first form in the document, and it can be linked to itself.

13. Select the **Skip attachments** check box to skip attachment forms that you want to exclude from the Implicit Form ID process.

14. Select **Save All** on the Designer window toolbar to save your changes.

Setting Virtual Form ID Properties

Form ID operates in the logical hierarchy of job definition and form definition. Virtual Classification (VC) extends that hierarchy to variant definition. Form ID uses VC by making use of multiple freely located fields of textual data in unique positions on a layout variant of a form type (e.g., invoice or claim) to determine sub-classes and variants.

Auto-Training or Manual Training create the variant definitions, which are used by Form ID, recognition engines and Completion to facilitate and accelerate OCR and key-from-image (KFI). If KFI or OCR data match data on the original image, Auto-Training can assign zone coordinates in the virtual forms file. KFI forms and fields lacking accurate OCR require Manual Training to add zones or to modify the zones set in Auto-training.

The sub-components of Virtual Classification include:

- Generic form
- Form sub-class
- Form variants
- Auto-Training
- Manual Training

Understanding Virtual Classification

Processing forms using Virtual Classification (VC) involves a generic form, form sub-classes, and form variants. It enables you to capture the same type of data (e.g., patient name, SSN, date) from forms with different layouts. Using VC, you set up a generic form to capture specific data fields from a form type, such as claims or invoices. You can have form sub-classes for each form type to designate form layout variations. For example, two forms from the same vendor would be in the same sub-class. Form sub-classes subdivide further into form variants, which have identical ID text strings (e.g., invoices from the same vendor), but which have differing layouts.

You can use Virtual Classification (VC) to identify and optimize recognition and key-from-image (KFI) processing of form variants. Instead of creating a new form definition for each form, VC creates, identifies and stores information in a set of variant definition files (rather than job definition and form definition files). Auto-Training or Manual Training creates the variant

---

definitions, which are used by recognition engines and Completion to facilitate and accelerate OCR and KFI. If KFI or OCR data match the original image data, Auto-Training can assign zone coordinates in the virtual forms file. KFI forms and fields lacking accurate OCR require Manual Training to add zones or modify the zones set in Auto-Training.

You must have the following FormWare licenses to use VC:

- VtrnClnt.exe – Virtual Classification Auto-Training Client
- Mantrain.vbp – Virtual Classification Manual Training VBA Project
- FIDVirtual Classification – Form ID Virtual Classification

Understanding Generic Forms

A generic form definition file (FDF) contains the data fields and data entry Completion screen layout that you need to capture data from any form presented for processing (e.g., invoice, purchase order, or change of address form). An invoice form may include many of different paper-form layouts. However, ultimately the data from all of these layouts will be mapped into a specific set of common data fields so that they can be captured and exported to a standard destination format. For each generic form there is a single FDF that includes:

A list of all common data fields to be captured (including those fields that will be used for VC).

The data properties of those data fields, including: field data type (e.g., date or amount), length, and common validation edits required to capture valid data.

Completion data-entry layout properties for those fields–where the fields will be displayed in the Completion data entry screens, what order the fields will be entered, and how the field-to-field movement and validation edits will be performed.

Free-form field configurations for every data field detailing all needed free-form information with an image zone that provides a general idea of where on the paper form the data is likely to reside. This may be as general as "the top ¼ of the page" or "the whole page," if the data are equally likely to appear any place on the page.

Note: A generic form should not contain mark sense nor handprint fields, because FormWare does not currently support free-form mark sense or handprint fields.

Starting with a generic form, you use Auto-Training and Manual Training to create sub-classes and variants used to process forms with many different layouts, orientations, and sizes that all have the same common data fields. In FormWare, a job can have only one generic FDF. All other information specific to different variants is stored and referenced in a set of virtual Form ID and virtual form data files associated with the FDF. This information is first used to identify the specific form sub-class and variant and then to dynamically modify the field zones for the specific image as needed by the FormWare client.

Understanding Form Sub-Classes

A form sub-class includes form variants that have a common set of VC field data. For example, if the VC data are a fax number and zip code on an invoice, that information uniquely identifies a specific vendor as a sub-class of the generic invoice FDF. The sub-class data file contains information that allows FormWare to identify unique layout variations of the generic base form. In most cases, a sub-class possesses unique Form ID text properties, corresponding to a unique form layout (variant). A generic form may have many sub-classes (e.g., user-defined variations, such as vendors). FormWare supports an unlimited number of sub-classes.

For example, an invoice from ABC Corporation would be a form sub-class distinct from an invoice from XYZ Corporation. A sub-class must have at least one VC field that varies from the

generic invoice (e.g., vendor name). Form sub-class data storage contains the information the Form ID client needs to identify a form sub-class and variant.

Understanding Form Variants

A form variant is an alternate form layout for a single sub-class because it has unique field-layout properties. The virtual form data file contains zone layout data for all variants. A vendor with two different invoice formats (with the same sub-class information) would have two variants. VC identifies a form by determining the sub-class and variant to which an image belongs. Most sub-classes will have a single form layout and only one variant. It is possible, however, that vendor ABC may have two or more form layouts, which are called variants. In InvoicePack, forms from different vendors and two forms from the same vendor with different layouts would be unique variants.

Example 1:

The sub-class ABC, variant 1 invoice places the shipping method in the upper left of the page; the sub-class ABC, variant 2 invoice places the shipping method in the lower left of the page. A variant defines the field zone information and other differentiating information.

VC detects variants based on the position of the VC fields on the form. In this example, the fax number or zip code must reside in a different location on the page for VC to detect a different variant of the same sub-class. If the VC field data and location are identical, then only one sub-class will be defined.

Example 2:

Invoice might be the generic form. A sub-class of invoice is the specific invoice from XYZ Company. The XYZ sub-class displays the zip code in the upper left corner, which Form ID has registered. XYZ changes printers and places the zip code in the upper right corner. Auto-Training designates the new invoice form as a variant for the XYZ sub-class invoice. VC can then identify it as a unique layout.

The Form ID client marks the form number in the BDF and adds a parameter for each image, describing the variant identified. The parameter format is FW_VID=<Form Variant ID number list>. These parameters can also be inserted by Completion, if a variant is manually selected during data entry, and by Auto-Training when a new sub-class is created.

Understanding the Virtual Classification Image Training Process

Training creates variant definitions. Off-line training is useful when you can select a set of training images that represents the highest volume of form variants. When this is possible, Manual Training can continue to fine tune virtual forms processing quickly and reliably, even if OCR quality is relatively poor. Each time an unknown form is encountered during production, a new virtual form definition is trained automatically. When subsequent similar layouts are processed, FormWare automatically and quickly locates the needed information quickly. This self-training system captures information about each new variant, storing the data elements in the variant definition data files.

New variant definitions can also be introduced to the system in a manual pre-production step. The benefit of training is that known form variants are automated, while untrained variants go through OCR and then are processed by directed KFI made more efficient by allowing the user to key from specific fields as opposed to an entire document. The keyed data are then automatically stored in the sub-class and variant data files for future Form ID. Variant definitions do not clutter the system with thousands of individual entries; instead, a sub-class manager manages a central

repository. The sub-class manager can import Variant ID information directly into this central repository from an external host system (e.g., accounts payable or other financial software).

Understanding Virtual Classification Files

The following file definitions are provided for easy reference in implementing VC:

- **<FormName>.fsc** - The primary sub-class data file
- **<FormName>.vfm** - The primary virtual form variant zones data file
- **<FormName>.fst** - Transaction files for updating the FSC file
- **<FormName>.vft** - Transaction files for updating the VFM file
- **<FormName>.fse** - Temporary "exclusive access" files for the FSC file
- **<FormName>.vfe** - Temporary "exclusive access" files for the VFM file
- **<FormName>.mtr** - The Manual Training Configuration file
- **ManTrain.vbp** - The Manual Training VBA project file
- **ManTrain.str** - ASCII strings used in message boxes in the Manual Training VBA project
- **Vtrnclnt.ocx** - Auto-Training program file
- **VfidAdmin.exe** - Virtual Form Administrator program

Creating a Virtual Forms Application

Use the following steps to create a virtual forms application that includes sub-classes:

1. Create a job with a single generic FDF and a Manual Training file (MTR). The FDF should include ID fields and data fields. To accommodate Manual Training the FDF must conform to the guidelines described in the Using Manual Training section—you must create an MTR file using a text editor.

2. Import a setup image. Draw generic-form zones to cover the range of locations in which data can be expected; they can be scaled by Recognition and Completion clients to fit any size production image.

3. Create new variant definition files for the generic form. Open the Sub-class Manager, which prompts you to create a sub-class and virtual form files unless they exist already for this generic form.

4. Specify the fields for Virtual Form ID. Within the Sub-class Manager specify the field names to be used for the Sub-class Name field and the Form ID fields.

5. Set the default properties. Specify the sub-class properties to be assigned to new sub-classes as they are created by Manual or Auto-Training.

6. Import available ID info from a source database such as an accounting system (or skip to step 7). Save time and increase accuracy by importing sub-class ID data. If ID string information for each sub-class is available from a source database at the destination system, import that information (e.g., tab-separated ASCII data or tab-separated ASCII data with field names).

7. OCR or KFI (key and zone the information) if ID strings are unavailable for import. If Form ID strings from a source database are unavailable, decide whether they must be OCR or key entered. Either manually key VC info into the sub-class manager, key VC info during Manual Training, or KFI/Reject Repair the VC info to be used by Auto-Training during production.

8. Configure Auto-Training Parameters. Return to Designer and configure default parameters for Auto-Training.

9. Design the forms-processing application. As for any other application, determine and configure all field properties, edits, exports, JobFlows, etc.

10. Decide a variant definition fine-tuning strategy. You can manually train certain high-volume forms and images off-line (i.e., pre-select the forms and consider manually training them before performing production data entry). Or, you can start scanning and processing without prior training; let FormWare define variants on the fly. If you have only a few hundred variations, Auto-Training may be unnecessary.

Creating a Sub-Class and Virtual Form Files

A form is processed as a "generic" form if it has the following VC files associated with the FDF:

- **Sub-class file** - <form name>.fsc
- **Virtual form file** - <form name>.vfm)

You create the initial FSC and VFM files in Designer. When no VC files exist for the FDF, Designer prompts you to create them. To configure a virtual form application, you first create an FSC file and specify which fields on the form are to be used for VC. You then determine whether to import the Form ID information from a source database, or whether it must be processed through Recognition (OCR) or keyed from an image. Finally, you configure default parameters for Auto-Training and newly trained sub-classes.

Note: Choose fields that are likely to result in excellent OCR and that in combination can uniquely identify the form sub-class. Bad OCR leads to failed Auto-Training and manual intervention to continue processing.

There are three ways to create sub-classes:

- Form Sub-class Manager
- Auto-Training
- Manual Training

Using the Form Sub-class Manager

To use VC you must create form sub-classes and variants that FormWare will scan for matching fields during Form ID. You can use the Form Sub-class Manager to:

- Create an initial Virtual forms database to store the sub-classes for a generic form.
- Create sub-classes by importing sub class definitions from a prepared text file (often downloaded from a corporate database).
- Create sub-classes by manually entering the information.
- View, edit, and delete existing sub-class information.

Note: You can also use the Auto-Trainer to find and create sub-classes from previously unknown variants.

Creating an Initial Virtual Forms Database

When you access the Form Sub-class Manager the first time for a generic form that does not have a database, an initial database is created automatically. Before you create sub-classes in the database, you need to specify a few things after the initial database is created.

---

1. Open a job in Designer.

2. Double-click the form name in the Job Browser to display the Form Properties dialog. The job must have only one form, the generic form.

3. Select the ID tab.

4. Select the **Sub-class Manager** button.

5. If you have not previously created FSC and VFM files, you receive a confirmation message to create these files. Select **Yes** on each dialog to create these files. The Form Sub-class Manager dialog displays.

6. In the Fields Used For Virtual Form ID area, select the **Select** button to choose fields for VC. The Select Fields for Virtual Form ID dialog displays. The highlighted fields are currently selected for use by VC. You must select at least one field and a sub-class name from the **Sub-class Name** field before you can exit this dialog.

    FormWare supports zoning and keying data for VC ID fields. VC uses these zones to detect variants. The zones selected within the field list indicate which data fields the ID strings must match. This is also used during Auto-Training to detect potential new variants.

7. To add a field to the VC Form ID process, select it. To remove a field, select a highlighted field to deselect it. Any changes you make on this dialog are global and affect all of the sub-classes in your job.

Caution: If you deselect a field that contains sub-class data, you will lose the data in that field for all previously trained sub-classes. Therefore, if you reselect that field, no information will exist in that field for the sub-classes in your system.

---

8. In the **Sub-class Name Field** field, select a field for Auto-Training to use to name new sub-classes. For example, if you select the Vendor ID field (the default for InvoicePack), Auto-Training uses the Completion data you typed into the Vendor ID field as the name of a new-sub-class when it detects an image type it has never seen before. If you do not select a name, Auto-Training creates unique names that may not be significant to your application.

Note: The field list on the Select Fields for Virtual Form ID dialog and the Sub-class Name Field list only display those fields that are non-label fields and those that have image clips.

---

9. Select **OK** to confirm your changes and return to the Form Sub-class Manager dialog.

10. Select the **Set Defaults** button to create default settings for sub-classes so that each time you create a new sub-class you will have fewer changes to make. You set these default values on the Sub-class Default Settings dialog.

11. Select **OK** to accept your settings.

12. On the Designer window, select **Save All** to save your settings in the job.

Importing Sub-classes

Select **Import** to import externally-prepared Form ID string information (e.g., from a mainframe database) and quickly define form sub-classes and a single, default form variant for each. The **Import** button displays a dialog that allows you to select the import method. Select **Import** to enable browsing for sub-class data to be imported. There are three source options: Sub-class file, tab-separated ASCII file with field names, and tab-separated ASCII file. You can import just form sub-class information (i.e., with no variants defined) or a full sub-class file set.

Manually Creating a Sub-class

Select **New** to add a new sub-class. See the steps in the Adding a Sub-class section for more information.

Using the Grid Display

Because thousands of variants are possible, FormWare implements a grid control on the Form Sub-class Manager dialog to display the existing sub-classes. Each row represents a sub-class. Each column on the grid represents a Form ID property. The Form ID field column contains the field's clip override. You can determine the order of the columns and which columns display by right-clicking on a column name and selecting or deselecting column names. The grid columns include:

- **Name** - Column title is the name of the form sub-class as designated in sub-class setup.
- **ID Strings** - String values for the fields used for virtual Form ID. If you are using Auto-Training, the longer ID string you use will yield better performance and accuracy.
- **Match Threshold** - The number of matches on ID strings required for FormWare to validate the Form ID (normally 2 or 3 will suffice).
- **ID Order, Set Sequentially as Variants are Created** - Priority in which VC uses a sub-class. VC searches the sub-classes against the current image for the matching ID from the lowest number to the highest. Two or more numbers can be the same. In that case, the numbers have equal priority and VC searches them in the order they are listed in the grid. ID Order numbers are the same as the **Sequence Numbers** on the Sub-class Properties dialog. To invoke the sequential order, deselect the **Prioritize sub-classes** check box in the Settings area of the Virtual Form Auto-training dialog. With sequential order enabled, virtual Form ID might take up valuable production time when it encounters an unidentifiable form, because it first goes through all the variants. In that case, use the ID order based on the number of forms identified as each sub-class.
- **ID Order, Based on Number of Forms Identified as each Sub-Class** - Rather than have the ID order set sequentially as new variants are created, you can optimize the virtual Form ID throughput by allowing the Form ID order to be re-arranged automatically based on the number of forms that are identified as each sub-class. This ID order is the default (i.e., the **Prioritize sub-classes** check box is checked). The result is to minimize the number of variants by displaying the most frequently seen variants at the top of the ID order column list. Additionally, during Form ID testing, you can reveal how long it will take to go through all the variants in the system. Optimizing the ID order does not, however, speed up Form ID for unidentifiable forms.
- **Variant Detect** - Method to use for variant detection.
- **Color ID** - Displays the red (B), green (G), and blue (B) background values if you use a color image, enable color processing, and train the sub-class. In all other cases, this column displays "None." Training color images writes the color values to the ID file for each image. VC later uses these values to identify images from the same sub-class. For example, the color ID column could display 235,344,127 as one color ID string.
- **Auto-Train** – Displays Yes or No to indicate whether FormWare should automatically train the form by adjusting zone coordinates in the auto-trainer.
- **Auto-Train Limit** - Limits the number of images you automatically train during a processing session. Type the number of images you want Auto-Training to examine before performing any zone adjustments on this sub-class during production. This is the same as the Stop Training after field on the Sub-class Properties dialog.
- **Created** - Date this sub-class was created.

- **Modified** - Date of last modification of the sub-class.
- **Last Used** - Date the sub-class was last used.

In the Grid Control area, you can select a column title to sort the grid by that data.

Finding an Existing Sub-class Entry

You can locate a specific sub-class using one of the following methods:

- By sub-class name. Scroll down the list until the desired name is found. (To sort the list by name, select the name column's heading.)
- By using the Find Virtual Form search methods.

To use the Find Virtual Form search method:

1. Open the Form Sub-class Manager dialog.

2. In the **Find Virtual Form** area select from the following options to search for a specific sub-class:

   a. Select the **By Number** check box to find a sub-class by number.

   b. Select the **By Text** check box to find a sub-class from a text string.

   c. In the blank field, type a number or text associated with the sub-class depending upon which type you selected.

   d. Select the **Find** button. The system highlights your sub-class in the grid. If you type data found in more than one sub-class, the system highlights the sub-class that is alphabetically or numerically first.

Viewing and Editing Sub-class Properties

Select **Properties** to view and modify the properties for the highlighted sub-class. The Sub-class Properties dialog displays. You can also view properties by double-clicking the sub-class row in the grid.

Activating a Variant for Other Designer Operations

Select **Set Active** to view the zone settings on the variant setup image in Designer when you select OK. Any change made to a variant field clip affects the variant database and not the generic FDF zones. Changes to variant zones are saved by the same actions that save the form. As for the variant setup image, you cannot browse for or set the scaling factors in Designer. These features will be grayed out. This tells Designer to use the selected form sub-class and form variant when displaying and editing field clips. If the highlighted sub-class represents a group of variants, then the Select Virtual Form dialog displays to allow a single selection. This button is only meaningful when the sub-class manager is started from Designer.

Deleting a Sub-class

Select **Delete** to remove the selected sub-class. Deleting a sub-class also deletes all the variants associated with the sub-class. You can also perform this function by right clicking on a sub-class line in the grid and selecting Delete.

Adding a Sub-class

From the Sub-class Manager, you can add new sub-classes and edit the properties of existing sub-classes. New sub-classes are often added via Manual or Auto-Training, or by importing a vendor list. But they can also be added here.

To add a sub-class:

1. On the Form Properties, ID tab, select the **Sub-class Manager** button to display the Form Sub-class Manager dialog, select the **New** button.

2. In the **Name** field, type a name for the new sub-class.

3. In the **Form ID String** column within the **Form ID Definition** area, type the ID string values. The **Field** column displays the name of the fields currently selected for VC to use in Form ID.

4. Select **Properties** in the **Variants** area to add and delete variants to and from the sub-class. The **Variants** field displays the currently assigned variant numbers for all variants in the current sub-class. The Variant Properties dialog displays. Select the **New** button for each variant you want to add, then select **OK**.

5. Select the **Enable** check box in the **Auto Train** area to enable Auto-Training during production processing for this sub-class.

6. In the **Stop Auto-training after** field, type the maximum number of images you want to train during production. Auto-Training adds time to your production processes; therefore, you should set this value to a number that will optimize your production process. Also, after a certain number of image examples have been reviewed, training may converge on a specific set of zone locations. Therefore, you may want to limit the number of images you automatically train during a processing session.

7. In the **Variant Detection** area, select one of the following options to enable or disable variant detection. The option you select displays in the **Variant Detect** column in the grid area on the Form Sub-class Manager dialog. Options are:

   - **None** - Disable Variant Detection. When VC identifies an image from a sub-class with multiple variants, it attaches a list of all the variants to the batch. You can then select the correct variant for the image in Completion. Select the None option for sub-classes where variant detection produces more false positives than you desire.

   - **Form ID Data Locations** - Enable Variant Detection. VC tries to identify the variant of each image based on the page locations of the identification strings. If you previously trained the locations of the identification strings, VC will identify the variant. In Completion, the system displays the identified variant instead of all variants in the sub-class.

8. Select the **Colors** check box to enable color identification for the current sub-class. When you select this option, the adjacent field displays the red (B), green (G), and blue (B) background values when you use a color image, enables color processing, and trains the sub-class. Training color images writes the color values to the identification file for each image. VC later uses these values to reduce the possible set of matching sub-classes.

9. In the **Match Threshold** field, type the number of Form ID fields that must match the stored sub-class match string to successfully ID that given sub-class. For example, a form has two ID fields: one with a match string of TEMPUS, the other with a zip code field that must match 23495. If the match threshold field is set to 1 and TEMPUS is recognized on an image, but OCR for the zip code is not correct, then the form will still be identified as TEMPUS. If the match threshold is set to 2, then both fields must match.

10. In the **Seq** field, type the sub-class sequence number. This value represents the numeric priority in which VC uses a sub-class. VC searches the sub-classes against the current image for the matching ID from the lowest number to highest. Two or more sub-classes can have the

same sequence number. If so, the numbers have equal priority, and VC searches them in the order they are listed in the grid.

11. Select **OK** to save your changes and return to the Form Sub-class Manager dialog.

Adding or Deleting Variant Properties

To add or delete a variant property:

1. From the Sub-class Properties dialog, select the **Properties** button in the **Variants** area. The Variant Properties dialog displays. The grid lists the virtual forms in the current group. The first column is the virtual form (variant) number. The remaining column headings are the names of each Form ID field. The numbers in each cell are that field's clip coordinates. The numbers under each clip represent location coordinates of the respective recognition zones for each field.

2. To add a new variant (virtual form), highlight a blank row and select the **New** button. This assigns the variant a unique number for a possible new virtual form and adds that number to the grid. The field zones are defaulted from the generic form. You can override these zones with manual adjustments in Designer by dragging the normal field-image clips during Completion.

3. To delete a sub-class, highlight the row and select the **Delete** button. This deletes the virtual form from the group. Because a virtual form must be part of some group, the form itself is also deleted. A sub-class group must have at least one variant (i.e., one virtual form). Hence, you cannot delete the last variant assigned to a sub-class.

4. Select **OK** to return to the Sub-class Properties dialog.

When Completion applies a variant to a form, it overwrites the form zone settings from Designer with the zones stored in the variant files. During Manual Training, when the variant does not have zone information for a given field, the form zone is automatically adjusted using the difference in size between the setup image and the current image.

When you set up Virtual Classification in Designer, you select a default setup image and associate form variants with that setup image. Whenever, Designer detects a change to the setup image, you will be prompted as to whether you want to store or update the value of the FW_VFRM_GOLD parameter.

Adding Variants for Non-Free Form Fields

To add a variant for a non-free form field:

1. The post-Completion data are matched against the Recognition data (either from the OCR file or re-Recognized full page). If the match fails, Form ID ignores the field on this image, and no adjustment occurs.

Note: When full-page orientation detection is turned on, only two orientation settings are applied during Recognition processing – normal (0 degrees rotation) and upside down (180 degrees rotation), as opposed to applying all four orientations (0, 90, 180, 270). This speeds up the full-page recognition-based Form ID by a factor of two.

---

2. A union of coordinates of all the instances of the field for the variant is built up, expanded by a fudge factor, and then compared to the existing variant field clip or to the clip on the generic form, if no variant clip exists.

3. If there was no pre-existing variant clip, then a new variant clip is added as the box-union clip

---

4. If a pre-existing variant clip is present, then Form ID adjusts it a percentage closer to the box-union clip—i.e., a percentage based on ratio of the number of the variant form's images in the input batches to the variant form's overall reference count. The is the Learn Rate.

Adding a Field for Virtual Classification

To set up a variant you must first select the desired form variant. Once the variant is selected, Designer automatically provides access to that variant's field clips and form IPP settings, instead of showing the items of the generic form.

1. On the Form Properties, ID tab, select the **Sub-class Manager** button to display the Form Sub-class Manager dialog with the current default clips highlighted.

2. Highlight the desired form variant on the grid and select the **Select** button to display the Select Fields used for Virtual Form ID dialog.

3. The highlighted fields are already part of the ID profile. Select the non-highlighted field you want to add and select **OK** to add it to VC fields.

4. Select **OK** to return to the Form Properties, ID tab.

The **Sub-class Name Field** list shows the fields that are on the base form. Because these are virtual fields they will never have a recognition zone. You select from this list to pick the fields used for this virtual form.

Removing a Field Definition

You can remove a field when it is no longer needed or you want to remove it to facilitate recognition or keying. To remove a field definition:

1. On the Form Properties, ID tab, select the **Sub-class Manager** button to open the Form Sub-class Manager dialog.

2. Highlight a field row.

3. Select the **Select** button to view the Select Fields used for VFID dialog, which reflects the defined fields for this form.

4. De-select the highlighted row that you want to delete and select **OK**.

5. On the confirmation dialog, select **Yes** to delete the field.

Note: If you delete or re-order one or more fields in Designer and save the new FDF, you must perform a database update using the VFIDAdmin.exe before you run any other JobFlow tasks, including Completion. This will ensure that the FDF is in sync with the new files.

Remember that when you add to the Sub-class Manager you need not update the files, but when you delete, you do have to update the files.

Importing Sub-class Data

For forms you intend to train manually, the most efficient method for designing a VC application is to import ID string information from a source database. By importing existing files containing sub-class identification you can quickly define form sub-classes and a single default form variant for each. Besides saving time, you will increase accuracy and ensure consistency with your destination system.

Importing sub-class data from a source database into a target database within VC is critically dependent on whether sub-classes already exist. Do not import sub-class data over existing sub-class data (i.e., same sub-class file names exist in both the target and source areas). Before

merging the databases, you must verify that the fields from the header of the source database exist in the FDF and are associated with the target database (e.g., ensure that the source and target databases were created using the same FDF). If they do not match, an error message displays indicating there is a mismatch and that the import cannot be performed. If the databases do match, you should check every record imported. Check the following:

- Whether the sub-class in the target database is the SAME NAME as the sub-class in the source database, OR
- Whether ALL of the Form ID fields AND all of their values (in the target database) exactly match ALL of the Form ID fields AND all of their values (in the source database).

When either of these is satisfied, a dialog displays that provides the following choices:

1. Keep the existing record in the target database

2. Replace the record in the target database with the one from the source database

3. Select the **Apply to all** check box to prevent this dialog from appearing multiple times.

When neither of options occurs, the import proceeds and displays a message when the process completes successfully.

FormWare can import just form sub-class information (i.e., no variant zones defined), or a full export. To import sub-class data:

1. On the Form Properties, ID tab, select the **Sub-class Manager** button to display the Form Sub-class Manager dialog.

2. Select the **Import** button to display the Import Sub-class Data dialog.

3. Select the file type you want to import from the following options:

   - **Sub-class File** - Import an existing FSC file. The default location for existing FSC files is the \FormWare\Forms folder.

   - **Tab-separated ASCII file with field names** - Import a tab-separated ASCII file (a Text Import Data file with a .imp extension) that contains field names. The default location for existing Text Import Data files is the \FormWare\Forms folder. When you select this option, you can choose to select the **Sub-class names are included in import file** check box to indicate whether the actual sub-class names exist in the import file or not.

   - **Tab-separated ASCII file** - Import a tab-separated ASCII file (a Text Import Data file with a .imp extension) that does not contain field names. The default location for existing Text Import Data files is the \FormWare\Forms folder. When you select this option, you can choose to select the **Sub-class names are included in import file** check box to indicate whether the actual sub-class names exist in the import file or not.

4. If the Text Import Data file you are importing contains the names of the sub-classes, select the **Sub-class names are included in import file** check box.

5. Select the ellipsis (…) button to browse for the file you want to import (that was exported from another system. The default location for existing files is the \FormWare\Forms folder.

6. Highlight the target file and select Open. The file name displays in the File field.

7. Select Start to see the imported sub-classes in the Form Sub-class Manager dialog.

Note: When you import new sub-class information into a production system, you must either discard the old database and load the new sub-class files or merge the new files with the existing files using the import feature. If there are conflicts, FormWare allows you to decide which sub-classes to keep by running a query.

Using Auto-Training

When you use Auto-Training, you will experience the advantage of directed KFI, faster OCR, more accurate OCR, and, in the case of invoice processing, vendor identification. The three main functions of Auto-Training are to:

- Create new form sub-classes and variants.
- Set ID order priority based on the frequency that certain sub-classes occur in production.
- Automatically update the virtual form zones for new and existing sub-class files.

Each time an unknown form is encountered during production, a new virtual form definition is trained automatically. When subsequent similar layouts are processed, FormWare automatically and quickly locates the needed information.

Note: You must run Auto-Training as an OCX process to gain maximum value. Auto-Training needs to maintain context across multiple batches, and that context is lost if the VTRNCLNT program is configured to run as an EXE.

Auto-Training runs in a JobFlow after Completion and VC. Sub-classes are created in the following sequence:

1. VC creates a virtual Form ID Data file and tags an entering image as a form variant or leaves it untagged.

2. An OCR process then optionally processes the image.

3. A Completion operator looks at the unknown form and identifies it as belonging to a form sub-class.

4. The Completion operator keys or corrects all field data.

5. Auto-Training (Vtrnclnt) compares the data that has been entered or verified in Completion to the full-page OCR (.fpo) file, looking for the data locations.

6. Auto-Training either creates new sub-classes or variants, or trains existing variant data zones by adjusting the field zones to allow for the location of newly keyed data.

7. Auto-Training then tags the images in the batch with the field numbers of fields that could not be automatically trained, because of poor OCR quality in the VC step. Poor recognition leads to limited or no Auto-Training. Auto-Training adds the following tags to the BDF for Manual Training:

   - FW_VID=variant# (Added when Auto-Training creates a variant for a previously untagged image. Create New Sub-classes is enabled.)
   - FW_VFLDS=#,#,#, (Added to indicate fields for which Auto-Training could not determine a zone for the variant.)

Performing Auto-Training

Use the Virtual Form Auto-Training dialog to enable FormWare to automatically train your application to recognize different form types and generate the necessary sub-classes and form variant information. Automated training is useful when you have a large variety of form types in your system. It allows setting up the system from scratch, and automatically adjusting the system to optimize performance over time.

When you use Auto-Training:

- Each time an unknown form is encountered, a new virtual form definition is trained automatically. When subsequent similar layouts are processed, FormWare quickly locates the needed information.
- As it sees new examples of a known virtual form, Auto-Training can adjust the known virtual form to account for the differences between it and the previous examples.
- Auto-Training can optimize Virtual Form ID so that an image is first checked to see if it is one of the most frequently occurring virtual forms, before checking to see if it is an infrequently used virtual form.

Before setting up Auto-Training, you must establish the following for the form:

- A Virtual Forms database for the generic FDF, including selecting the fields to use for Virtual Form ID
- Form ID to recognize a generic FDF
- Recognition (with no variant defined yet)
- Completion (blank data form for new variant)
- KFI (to enter vendor ID and variant data)
- Reject Repair on the generic form

To set up Auto-Training:

1. Open the job in Designer.

2. Double-click on the form you want to Auto-Train to display the Form Properties, General tab.

3. In the **Component** field, select the **Virtual Form Auto-Training** option and then select the **Configure** button. The Virtual Form Auto-Training dialog displays.

4. Select the **Create Sub-classes and Variants** check box in the **Settings** area to direct Auto-Training to detect whether unidentified images are new sub-classes or variants. FormWare adds a detected new sub-class or new variant to the database. If an image has not been previously identified as a variant, Auto-Training assesses whether the data entered in the string ID fields match an existing sub-class. Barring a match, the image belongs to a new sub-class.

5. In the **Examples Required** field, type the number of images in a newly detected sub-class Auto-Training must see to add the sub-class and its initial variant to the database. The more examples you designate, the more apt you are to prevent rarely seen invoices (or typos in the sub-class string ID fields) from creating spurious database entries. The default is 1.

6. Leave the **Prioritize Sub-classes** check box in the Settings area selected. This causes FormWare to rearrange the Form ID order when you use Auto-Training in the WorkFlow. The rearranged ID order occurs automatically based on the number of forms that have been identified in each sub-class. This default selection also assigns a priority to the created sub-classes. ID prioritization can improve the efficiency of VC by prioritizing high-volume sub-classes ahead of low-frequency sub-classes. Set ID order priority based on the frequency with which certain sub-classes occur within your production environment.

Note: If you deselect the **Prioritize Sub-classes** check box, FormWare sets the ID Order sequentially as variants are created. Virtual classification searches the sub-classes against the current image for the matching ID from the lowest number to the highest. See the Using the Form Sub-class Manager section for details on arranging the ID order.

7. Select the **Adjust Variant Field Zones** check box to direct Auto-Training to optionally expand the field locations of known variants based on new examples of those variants. When a new virtual form is created, Auto-Training sets the field locations based on where the data on the variant's initial images were located. Additional field clips may be set for the variant during Manual training, where you see a single image and draw field locations to cover the data.

Note: Both of these techniques for setting the field locations may not make the clip big enough to cover data that shifts around between instances of the same variant, such as Free Form data. Selecting the Adjust Variant Field Zones check box allows the Auto-Trainer to examine the subsequent examples of a variant and expand the field locations to account for these different locations of data.

---

8. In the **Age** field, type the number of days that designates the sub-class as lower priority if it has not been processed in this time frame. Sub-classes that have been processed within the number of days specified are assigned a higher priority than those that have not been processed for more than the age-designated. Use this value to prevent old, unused sub-classes from getting a high priority. The default is 180 days.

9. In the **Images Required** field, type the number of images required before a sub-class receives a different priority. Setting this value to a larger number allows prioritizing to occur only when Auto-Training has seen enough images to make re-prioritizing worthwhile. The default is 100.

10. In the **Examples Required** field, type the number of processed-image examples of a variant you need before recording adjustments to the database and reporting un-adjusted zones for Manual Training. A larger number of examples allow more images to be examined before determining that a zone is "not adjustable" and therefore reported as needing Manual Training. The default is 1.

11. In the **Log File** field, type the name of the log file for Auto-Training. This value defaults to c:\AutoTune.log. If this value is set, then Auto-Training outputs a summary of the database additions and adjustments it has made. The log file is always appended to, so you must periodically delete it.

12. In the **Test** area, select the **Select** button to browse for a batch to test. The batch must have already been processed up to the Reject Repair task. Select an input batch and then a testing task from among the three options. After the test, use the **View Log** button to see a summary of the test results.

13. Select the function you want to perform from the following options:

    - **Create Sub-classes and Variants**
    - Prioritize Sub-classes
    - Adjust Variant Field Zones
    - Save Test Results to Data Files - Manually controls additions and changes to the database (especially if you decide not to run the Auto-Trainer as an automatic JobFlow task). After a Test, you can save the test's sub-class and variant additions and modifications directly to the database. If you exit the dialog without saving the test results, any results from tests will not be saved in the Virtual Classification database.
    - View Log - Displays the contents of the log file.

14. Select **OK** to launch the test and, alternatively, view the test data.

15. Return to Designer and access the Form Sub-class Manager dialog by selecting the **Sub-class Manager** button on the Form Properties, ID tab.

---

16. View the new sub-class and the fields that it trained on.

17. Run a production run to auto-train a set of images. Auto-Training trains on the fields it can. Verify the Auto-Training process during the production run.

18. When Auto-Training has finished, perform a Manual Training step in Completion Remove Flags mode. Look at fields on which Auto-Training failed to train (indicated by the Draw Me message in each failed field on the data form).

19. Perform manual zoning on the Draw Me blanks.

20. Select **Save All** on the Designer window toolbar to save your changes to the form.

Auto-Training Variant Definitions

To auto-train the variant definitions, you can insert a base-client-derived .ocx (VTrnClnt.ocx) into a standard FormWare JobFlow after the Completion step. VTrnClnt.ocx is based on keyed data from Completion (via BCH output files) and a comparison of that data to the output of the Form ID client. This output is retrieved from the Form ID client via a new set of files containing detailed OCR data and character coordinates (FPO) files. Auto-Training also outputs a log file containing information on modifications to the database that were made during training. This client's processing options are configurable using Designer.

Auto-Training Color Images

Auto-Training reads the background color tag information set by the color IPP module and uses it to train ID color values for each new sub-class. For each sub-class, Auto-Training stores color values in the VC data files. Color values can be used to narrow down the search space when performing VC.

The Form ID client can use information output from FormWare Color IPP. Through color IPP two processes are at work in Form ID.

- **Prior to Form ID**. Auto-Training trains an image with background color settings with which IPP tags an image.

- **Color pre-ID**. Form ID reads the color pre-ID settings and attempts to classify an untagged image (no Form ID and variant ID number) in the input data item. Form ID finds the background color tag for an image and checks it against existing color values in the sub-class database. All matches within the threshold defined on the IPP Properties dialog, Color tab are deemed successful. This limits the number of sub-classes that must be tested for ID string matches, and can increase both the accuracy and speed of VC.

To implement color background detection and removal:

1. Open the job in Designer

2. Display the Job Properties, IPP tab and select the **Properties** button. The Image Pre-Processing Options dialog displays.

3. Select the **Color Processing** option and then the **Properties** button. The Image Pre-Processing Properties dialog displays.

4. Display the Color tab. See the Setting Color Properties section for more information.

5. Select the **Enable Virtual Form ID Tags** check box. The Form ID and Auto-Training components write the background value to a BDF file for future use. The Red, Green, Blue, and radius values are written out as a single property. For example:

- **JOBNAME=Color** - The input image color1.jpg (converted to color1.tif)

- **Background values** - Three-part background color value for VC (red = 22, green = 125, and blue = 34)

- **C:\formware\images\color1.tif** - Path to the resulting TIFF

Any processing of the binary image (such as deskew or cropping) is also written to the BDF, so that Completion can properly display color images that correspond to OCR data from the binary images.

You can adjust the sensitivity of the Color Form ID process by adding the FW_COLOR_THRESH= parameter to the FIDClnt.ini file. The valid range for this parameter is 0 –255 (units are in brightness). A value of 0 means that the background color of the input image must exactly match the database-stored "GOLD" sub-class RGB value for that image to be classified as that sub-class. Larger values allow for a range of brightness away from the gold value for successful Form ID. The default (recommended) value is 10.

Using Manual Training

Manual Training is performed by Completion using the ManTrain.vbp VBA project. To manually train a new form using Completion in VF Training mode, set up a Completion New step in the JobFlow that you can rename Manual Training. During Completion New, operators can key Form ID fields, manually zone the form data fields, and enable FormWare to train that form. Manual Training must be performed on images that have undergone the IPP and image rotation that will be performed in production. It is essential to have images that have been created on the production scanner, in the same resolution and with the same image placement and borders as will be present during production.

Manual Training can be used in two ways:

- As an off-line training function, common forms can be pre-selected and processed through a JobFlow that performs IPP and Form ID, then a Completion/New -Manual Training step. This operation allows you to choose a set of common form types (images) and create sub-classes and virtual form zones manually for common form types.

- Manual Training can also be run following Auto-Training in Remove Flags mode to train the fields that could not be auto-trained. After Auto-Training has trained the fields, Manual Training enables you to add virtual form zones for the remaining fields.

To perform manual training:

1. Create a generic FDF and an empty sub-class file with properly configured VC fields.

2. Select a set of training images that represents the most common form variants. Find the 20% of images that are expected to make up 80% of your processing volume.

3. Process the images through IPP (e.g., crop and deskew) and through Form ID (to possibly rotate landscape documents).

4. Process the images through Completion - New mode with the ManTrain.vbp.

5. View your new sub-class file in the Sub-class Manager. Use the Virtual Forms Administrator to update the data file (including all transaction files generated by training) into the primary data file.

6. Run a test production WorkFlow to determine the effectiveness of training.

7. View the results in Completion.

Prerequisite Files for Manual Training

Manual Training requires that the fields on a form can be classified into functions as defined in a Manual Training file (MTR). The chief prerequisite for Manual Training is a form layout that has fields that map directly to a sub-class and variant data zones. The top portion of the form must also contain a separate set of VC fields used for ID and the bottom portion must contain data fields. A FIDOK field follows the ID fields, and a DATAOK field follows the data fields. The ManTrain.vbp interprets the MTR file during Completion and provides edits to control the quick zoning of fields. It then stores the information into a pre-existing sub-class and virtual form zone file. To do this, the ManTrain.vbp reads two files:

- **ManTrain.str** - Located in the \FormWare\VBA folder, which all other ASCII strings use in messages from within the VBP project. These are stored in a separate file to facilitate localization.

- **Manual Training file (.mtr)** - Includes information on the function of certain fields in the generic FDF. Also includes ASCII strings used for messages, which can be changed or translated by the creator of the new application. Mantrain.vbp reads the MTR file, which describes the function of each field during training.

Creating a Manual Training File

The MTR file includes six quoted strings per line. The six strings include:

- If the first string equals $FSPEC, it is a field specification.
- The second string is the field name.
- The third string is the field type. Field types include:
  - **FIDSELECT** - The sub-class name field. Your generic form must contain one FIDSELECT field, and it must have an image zone. This field will be auto-populated with the name of a selected sub-class, or it can be entered to create a new sub-class name.
  - **FIDVARIANT** - Field that displays the variant number when selected. This field is never keyed, but is used only to display a variant number.
  - **FID** - VFID data fields that contain the Form ID strings. These fields must have image zones and may be recognition fields also.
  - **FIDOK** - A field that can accept three values, Y, N, or E. Use this field to enter whether the Form ID fields entered are OK (Y), or need to be re-entered (N), or should be edited (E).
  - **FIRSTDATA** - First data zone on the form
  - **DATA** - All subsequent data fields that require zoning.

Note: Column fields require only the column header to be manually zoned, and US address blocks require only one field to be zoned as a block.

  - **DATAOK** - The final OK to accept the form for adding to the sub-class data files (values are Y or N).
- The fourth string is text to display before field is processed.
- The fifth string is text to display after field is released.
- The sixth string specifies dependent fields, which include:
  - An empty string, indicating no dependent fields.
  - A list of field names that should receive the same image zone as this field (useful for address blocks that will be parsed by recognition edits into separate fields).

      o   A "$" followed by a number "X," intended for stretching and copying the same zone info from a single zoned header field to all rows in a column. For this option to be used, fields must be named ending with an underscore then a row number (e.g., LineQuantity_1). Then other fields named LineQuantity_<n>, for n=1 to X, will be set to the same zone coordinates.

If first string equals $SCALE, this line in the file is a copy of your image scale settings from the FDF. You must use this setting if the FDF scaling is not 100%. Then, the second string equals the X-scale, and the third string equals the Y-scale.

To create your own FDF that will be manually trained you first need to create the MTR file.

Caution: Only a trained FormWare administrator should attempt to create the MTR file. Successful creation demands extensive awareness of field types, names, and specifications.

1. Open MS Notepad or other available text editor.

2. Type the strings the strings listed above.

3. Name the file with a .mtr extension.

4. Save and close the file.

Performing Manual Training

You perform Manual Training with the ManTrain.vbp VBA project during Completion. Manual Training requires that the fields on a form be classified into the functions defined in the MTR file. To train your files and update them for use with Form ID, InScript, or Completion:

1. Ensure that a FSC file and VFM file exist for your base FDF. (These files exist already for InvoicePack). Otherwise, create them according to the directions in the Creating a Form Sub-class section.

2. Create a BDF that contains the images you will use to train the system. There are two methods:

    *   (BDF without Form ID numbers and FW_VID tags) during training select or enter a sub-class name and variant number for each image. The large image zones from the base FDF display. If you train on an existing sub-class, the image zones from the VFM file display.

    *   (BDF with ID numbers and FW_VID tags) the Form ID fields are automatically populated from the database. If necessary during training, you can select or type a sub-class name and variant number for each image.

3. On a command line, type the following command: **Complete –e –VBP=MANTRAIN <bdfname>**

    Use the options on the Virtual Form Administrator dialog to update the VC internal file structure with new sub-class and form variant information. The VC internal file structure follows the same file system hierarchy of FormWare; all design properties are based on the Job > Form file hierarchy. In addition, VC uses a new file type – variant definition files – to classify and organize multiple form layouts into sub-class and form variant information within the VC system. There are two basic types of Variant Definition files: transaction files (i.e., FST and VFT files) and database files (i.e., FSC and VFM files). The information for sub-classes and form variants is generated when you manually or automatically train your FormWare application to use Virtual Form ID. This new data is stored in the Variant Definition (transaction) files and is only updated when you deliberately update this information using the Update function on the Virtual Form Administrator dialog.

Caution: Do not use the update functions on the Virtual Form Administrator dialog in your Production environment while it is running.

4. In the **Vendor ID** field, type the sub-class name. (You can also press [F12] to display the Change Variant menu and select an existing sub-class name.)

5. In the **Vendor Variant** field, type a 1.

6. After entering or verifying the selected Form ID info, and zoning all ID fields, type Y in the OK? field in the Form ID form area.

7. Use the mouse to draw zones in each field. When you draw zones, include the keyword (e.g., "invoice date") AND the data (Jun 14, 2002), plus an additional ¼-inch border. The wider border accounts for the registration variability likely to be caused by scanner variations. Arrow back and forward to see previous fields. If necessary, re-zone them with required zones and borders.

Caution: The more keywords you assign and designate the more adjustment performance suffers. Exercise caution in adding keywords so as to minimize the effect on performance.

8. Draw the block fields for address components in the setup image sector.

9. For columns, draw the header, which consists of the keyword and the first data row.

10. After drawing the fields and zones, you can arrow back to ensure that you are satisfied with your zones. When you are satisfied, type Y in the **OK?** field. The VC files will be updated via Transaction files (i.e., <form name>.fst and <form name>.vft).

11. Continue with the rest of the batch.

12. Press **[Shift-F5]** to skip any form and not train on it at all. A message asking if you want to skip the zone adjustment step for this image displays.

13. Select **Yes** to skip the form or **No** to return to the form and continue Manual Training.

14. When you are finished, exit Completion.

15. Update the VC files with the new training data using the Virtual Form Administrator. From Administration, select **Tools > Virtual Forms Administration**. The Virtual Form Administrator displays.

16. In the **Sub-class File Name** field, select the **Browse** button and find the FSC file in the \FormWare\Forms folder. This file will have the name of the form (or job) that you just modified with the Manual Training process.

17. In the **Select Operation** area, select from the following options:

*   **Update** - Updates the VC base files by incorporating all information from the new transaction files. This moves information obtained through training from the transaction (FST and VFT) files to the data (FSC and VFM) files.. The transaction file format differs from the data file format. The transaction file contains Modify, Add, and Delete records. These records tell FormWare how to update the data files with the correct data. FormWare then empties the transaction files after each update.

Note: Captiva recommends that you back up your variant definition files daily and perform a system update after any substantial period of Auto-Training or Manual Training session.

*   **Roll Back** - Removes information from the transaction files and the data files by cleaning out transaction information back to the date and time entered in the **Date** fields.

- **Crash Recovery** - Deletes all transaction information from your transaction files. This does not affect the data files. Perform this task if your transaction files become corrupted

- **Recover From Exclusive Mode** - Removes the locks on the data files that FormWare uses when running in Exclusive Mode. This changes the E <Computer Name> lines to be R <ComputerName>, thus unlocking the files. Use this option when you get a locking error running in Exclusive Mode.

18. Select **OK** to accept your settings and return to the Virtual Form Administrator dialog.

Creating Virtual Field Zones

You can create virtual field zones in three different ways:

- Using Designer
- Using Auto-Training
- Using Manual Training

Creating Virtual Field Zones in Designer

In Designer you can also create virtual field zones for a sub-class, which VC uses to detect variants. FormWare supports zoning and keying data for VC Form ID fields.

1. Select the **Set Active** button on the Form Sub-class Manager dialog to launch the setup image so you can look at the field clips.

2. As needed, return to Designer and modify the field zones.

Note: The Designer-created or -modified zones are stored in the virtual forms file whenever you exit Designer by selecting the Save All button.

Creating Virtual Field Zones with Auto-Training

Use Auto-Training to create new sub-classes and optimize performance on an invoice-processing application that may include as many as 50,000 variations. Auto-Training runs during production in a JobFlow as a post-Completion task and automatically creates virtual form zones (also called variant zones) and possible sub-classes by comparing keyed data to location data created by an earlier VC step. In general, Auto-Training trains only those fields where both the keywords and the data have undergone accurate OCR. The virtual recognition client uses the full-page-OCR files that Form ID creates. If key-from-image (KFI) or OCR data matches data on the original image, Auto-Training can assign zone coordinates in the virtual forms file. Auto-Training specifies which fields need Manual Training by writing an image property that points out the numbers of those fields with poor OCR results. For example: FW_VFLDS=13,26,47

Creating Virtual Field Zones with Manual Training

Some forms or portions of forms, must be trained manually. Manual Training is a high-speed process for quickly adding sub-classes and virtual field zones or for reviewing existing sub-class zones. Fields and forms for which accurate OCR has not occurred require Manual Training to add zones for fields that Auto-Training did not add and possibly to modify the zones set in Auto-Training. Forms used for Manual Training must include the following sequence of fields:

- Sub-class name
- Variant number
- A number of Form ID fields

- Form ID OK field
- A number of data fields.
- Data fields OK field

These fields are described in an ASCII file named after the base FDF for the form and using the MTR extension (Manual Training file). Completion uses a VBA program to perform Manual Training.

Modifying Function Keys for Form Variants in Completion

By default Completion supports the following function key changes for form variants:

- **F11** - Displays the Change Form dialog.
- **F12** - Displays the Change Variant dialog.
- **Key 380** - Displays the Change Variant dialog from a hot key.
- **Key 381** - Toggles between multiple variants listed in the FW_VID property from a hot key.
- **Key 382** - Displays the Find Variant function from a hot key.
- **Key 967** - Displays the field clips in the attachment viewer if it is active for the application.

You can change these keys as necessary for your Completion process.

Cleaning Up Virtual Form ID Data Files

Once processing is complete, you can clean up the VC data files for a job from your system. To set up this process:

1. Open the job in Designer.

2. Display the Job Properties dialog.

3. Select the Cleanup tab.

4. Select the Virtual Form ID Data Files check box.

5. Save the job.

Testing Virtual Form ID

You can select the recognition engine to use for Virtual Form ID processing for a job. To run this test:

1. Open the job in Designer.

2. Open the Job Properties dialog.

3. Select the **Test** tab.

4. In the **Virtual Form ID Engine** field, select: **InScript**, **InScript2** (default), or **RecoStar**.

5. Select the **Test** button to run the test for the job.

When you close out of a job that uses Virtual Classification, the Auto-Training component will write information to a log file on any candidates that did not meet the repeat-count thresholds for sub-class creation.

Working with Form Variants in Completion

The functionality in Completion assumes that any job with form variant logic will have the Adjust Field Zones property set (on the Job Properties, Image tab n Designer). This is the sub-system used by Completion to track and store unique field clips per image. Completion also assumes that FSC and VFM files exist for each form that requires form variant support.

Completion provides the following support for form variants:

- **FW_VID=#[,#,#] (variant ID) property in the BDF**. When an image in the BDF is preceded by a FW_VID parameter, Completion will attach the first variant in the list to the first form loaded for that image. The form name on the status bar will have the variant number appended to it, as in "FormName (Var#)."

- **Function key to toggle between multiple variants from the BDF**. When the FW_VID property contains more than one variant, the first one gets attached to a form when it is loaded. After that, users can press the new Next Variant function key (381) to toggle through the variants in the list. When the end has been reached, it moves back to the first. The status line is updated to show the variant. The form name on the status bar updates to show the variant number.

- **Override generic field clips with the clips from a selected variant**. When a variant is loaded automatically by the FW_VID property, or changed by the Next Variant function key, or a variant is manually selected on the Change Variant dialog, the field clips from the selected variant override those from the generic form. You can reset the generic form clips on the Change Form dialog or by selecting 'None' on the Change Variant dialog. Both of these functions remove the variant number from the status bar. You can reset the generic form clips by re-selecting the current form from the Change Form dialog, or by selecting 'None' from the Change Variant dialog.

Note: Selecting "None" on the Change Variant dialog indicates that this image does not use a Form Variant. When you select this option, the associated "FW_VID" property tag is removed from the BDF file.

- **Auto-scale the field clips based on the variant specifications**. When you select a form variant, and the field clips are loaded, the scale factor for the generic form is applied to the clips.

- **Add a Manual Variant Select dialog**. You can manually select variants on the Change Variant dialog. This function supports a 'None' selection to return to the generic forms field clips. The status bar will update to reflect the new variant. The form name on the status bar updates to reflect the new variant number.

- **Variant search function**. If the wrong variant is specified in the FW_VID property, and you want to search for the correct one, you can enter data into the fields used for searching (criteria are determined when the FSC file is created) and select the Find Variant from the search menu or invoke the new Find Variant function (382).

Setting Manual Form ID Properties

Manual Form ID requires the Completion operator to manually determine which image corresponds to which data form during a Completion session. This manual process is used when the Automatic Form ID processes have failed to identify the image. During the Manual Form ID process, the Completion operator is presented with the unidentified images and types the correct form number for each image. This form number is then written to the BDF to associate the image with the data form.

Completion operators perform Manual Form ID during either the Image QC/Rescan mode or the Identify Images mode of Completion to determine which production image corresponds to which data form.

You can use the JobFlow Client to route batches that fail Automatic Form ID to the Identify Images mode of Completion so an operator can manually identify them. The operator can then route the batch to a Recognition or Reject Repair task for further processing.

Understanding Manual Form ID Menu (IDM) Files

When you design a job that contains more than one type of form, a Manual ID Menu file (IDM) is created to store that information. The IDM file contains a list of the form names within a job and an associated form number. This IDM file has the same name as the job with a .idm extension and is located in the \FormWare\Jobs folder. You can customize this IDM file for each job and then display it during a Completion session so that Completion operators can associate Form IDs to unidentified forms during the Manual Form ID process. IDM files are typically associated with a particular job, but you can create unique IDM files and run them exclusively with a specific Completion task by adding the -IDM=<IDM filename> parameter as a client parameter for the Completion task.

The basic IDM file has the following format:

```
Form ID number:form name
Form ID number:form name
Form ID number:form name
...
```

An example of an *unmodified* IDM file can look like this:

```
01:Form1
02:Form2
03:Form3
...
```

An example of a *modified* IDM file can look like this:

```
01:1 - 1040EZ Form 2002
02:2 - Schedule A 2002
03:3 - Schedule B 2002
...
```

Modifying an IDM File for a Job

Displaying the IDM menu within a Completion session enables the operator to associate a valid form name and number with an unidentified image. IDM files can be used within any mode of Completion but are most common in the Identify Images mode (Manual Form ID) and the Image Quality Check\Rescan mode (Manual IQC).

Before you modify any IDM file, see the Selecting a Display Format for the IDM File in Completion section to determine the best way to display the IDM file within a Completion session. Then you can perform the steps here to modify the IDM fie to suit the needs of your job or task.

To view or modify the Manual ID Menu file:

1. Open the job in Designer.

2. Display the Job Properties, General tab.

3. Select the **Manual ID Menu** button. The IDM file displays in a text editor.

4. Add to or modify the list of Manual Form ID options you want to display in the Assign Form (Change Form) dialog or a custom form menu during Completion. Each menu option value (Form ID number and name) must be on its own line followed by a carriage return. You can add the following parameters to the IDM file to change the display functionality of the Manual ID Menu:

VWR=#,#,#,#,#,#

This parameter enables the attachment viewer in the Image QC\Rescan mode of Completion (Manual IQC). The VWR (viewer) property overrides the Attachment Viewer properties that are normally set on the Job Properties > Image tab. The override is necessary, because Attachment Viewer settings for Manual ID/IQC typically differ from the settings specified for the job. When used in an IDM file, the VWR parameter must be the first line in the file.

Use the following values to define a full image view and specify a screen split in Identify Images mode of Completion:

```
VWR=520,0,50,0,1,0
```

Comma delimited values to the right of the equal sign are as follows (from left to right):

- o **Attachment viewer width** – Width (in pixels) of the attachment viewer window in pixels. Maximum width for any attachment viewer window is 700 pixels.
- o **Attachment viewer height** – Height (in pixels) for the attachment viewer to use to display images when the Switch Views feature is used. Default is 0.
- o **Initial scale** – Scale factor (percentage) to use for the attachment viewer image. When the scale factor is set, the on-screen image becomes x% of the original image.
- o **Initial rotation** – Initial rotation (degrees) of the attachment viewer window. Valid values are: 0 (no rotation), 90 (degrees), 180 (degrees), 270 (degrees).
- o **Scale to gray** – Set to 1 to enable scale-to-gray feature for attachment viewer images.
- o **Scroll** – Amount of pixels to scroll the attachment viewer image. Default is 0, 100 pixels is the maximum.

FWN=#,#,#,#

This parameter enables display of the Change Form dialog. Modify values to set the Change Form dialog position at the specified X and Y coordinates and have the specified width and height over the Completion window. Increasing the default size values makes more selections viewable within the dialog. Use these values to place the Change Form dialog over the top of the image thumbnail (default):

```
FWN=-1,0,0,0
```

Comma delimited values to the right of the equal sign are as follows (from left to right):

- o **X coordinate** – X coordinate position of the Change Form dialog.
- o **Y coordinate** – Y coordinate position of the Change Form dialog.
- o **Width** – Width of the Change Form dialog.
- o **Height** – Height of the Change Form dialog.

+/-Folder Levels/Names, Form Numbers/Names

Indicates the names and levels of folders and the numbers and names of forms within each folder. See the Modifying the IDM File to Use a Folder Structure task for instructions.

5. Select **Save** to save your changes.

6. Select **File > Exit** to return to Designer.

7. Select **Save All** on the Designer window to save your changes.

Modifying the IDM File to Use a Folder Structure
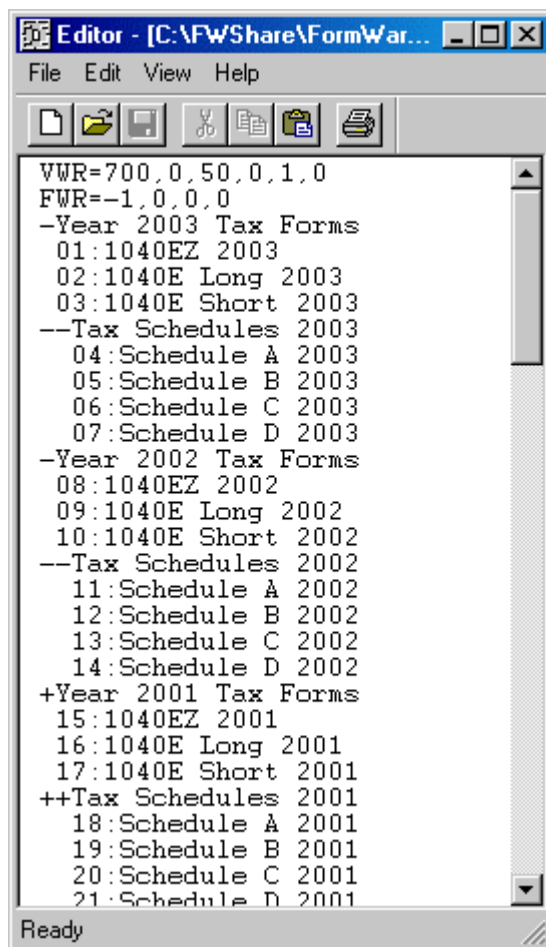
Some display options for Manual Form ID menus (IDM files) support the use of an expanded IDM file format, which enables you to create a hierarchical folder structure within the IDM file. This feature can be very useful when you have a large number of forms within a single job and need to classify them into smaller sub-groupings for ease of use.

By starting a line in the IDM file with + or –, you can designate that line as a folder name containing the form names and numbers immediately below it. Up to four folder levels are supported (++++ or – – – –) to organize forms within the IDM file into a multi-level tree view of folders up to four levels deep.

Note: You cannot have breaks in the folder structure (e.g., cannot jump from a level 1 folder to a level 3 folder)

When you insert + signs in front of a folder level name in an IDM file, it presents that folder as collapsed on the initial view. When you insert – signs, that folder defaults to expanded on initial view.

Here is an example of an IDM file modified to use a folder structure:

Here is what that IDM file looks like in the Assign Form dialog:
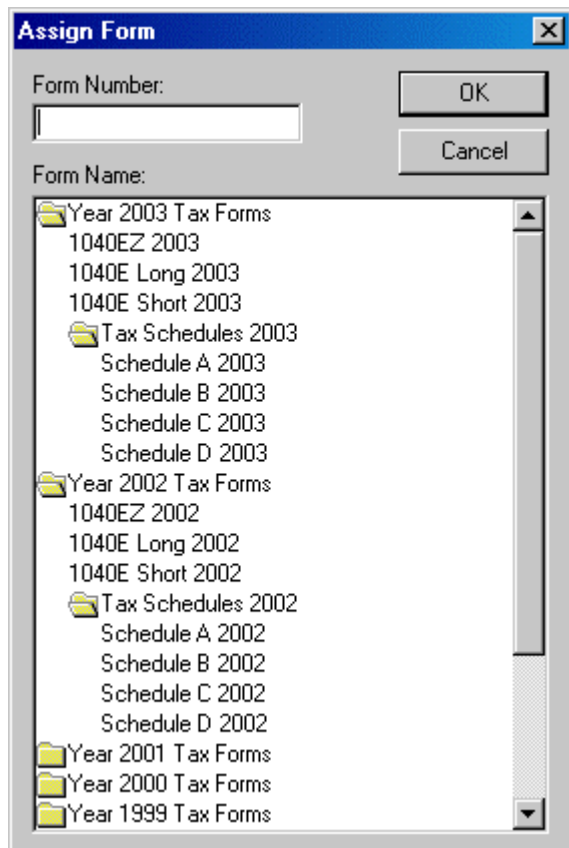
The folder structure on the Assign Form dialog enables Completion operators to double-click on a form description to assign Form ID to an image. As the operator processes images within the Completion session, the program saves the position of the last choice on the menu so that when a new image is presented, the cursor remains at the last form selection. The operator can also type a number for the form in the **Form Number** field. As the number is typed, the cursor moves to that location in the menu and highlights that form description. The operator can also type a form name and the program will navigate to that form name in the menu. Note that typing a form description is case sensitive.

Follow these steps to create a folder structure within the IDM file:

1. Open the job in Designer that contains multiple forms.

2. Display **Job Properties > General > Manual ID Menu**. The IDM file for the job displays in a text editor.

Note: For more information on the other parameter values in the IDM file, see the Modifying an IDM File for a Job task.

3. For a first level folder, type a single + sign and the folder name as you want it to display. For example:

```
-Year 2003 Tax Forms
```

Will display "Year 2003 Tax Forms" as an expanded folder on the initial view of the dialog.

4. For a form within the first level folder, type the Form ID number and the form name. Separate these values with a colon. Make sure to left justify the form ID and name with the folder name in the IDM file or they will not line up in the dialog. For example:

```
-Year 2003 Tax Forms
 01:1040EZ 2003
```

Will display form name "1040EZ 2003" within the folder "Year 2003 Tax Forms." Note that the form ID number and name are aligned with the folder name above it.

Note: As a good design practice, descriptions for the folder and form names should not start with a two-digit numeric value or they can be confused with the actual Form ID number.

---

5.   Select **Save** to save your changes.

6.   Select **File > Exit** to return to Designer.

7.   Select **Save All** on the Designer window to save your changes.

Selecting a Display Format for the IDM File in Completion

Once you decide to incorporate an IDM file in the Manual ID process, you should then choose the mode of Completion you want to use and how you want the contents of the IDM file to display to the Completion operator. You have the following display options for the Manual Form ID menu (IDM file):

Display Option One

Display Option One uses the basic Identify Images mode of Completion (Manual ID) and does not initially display a Manual Form ID menu to the Completion operator. During the Completion session, a single Form ID field displays where the operator can type the appropriate Form ID number that corresponds to the current image. The operator can also press [F5]-[F9] to display the Change Form dialog (IDM file), where he can either type a valid Form ID number or select the appropriate form name in the list. See the Configuring Manual ID to Use Display Option One task for instructions.
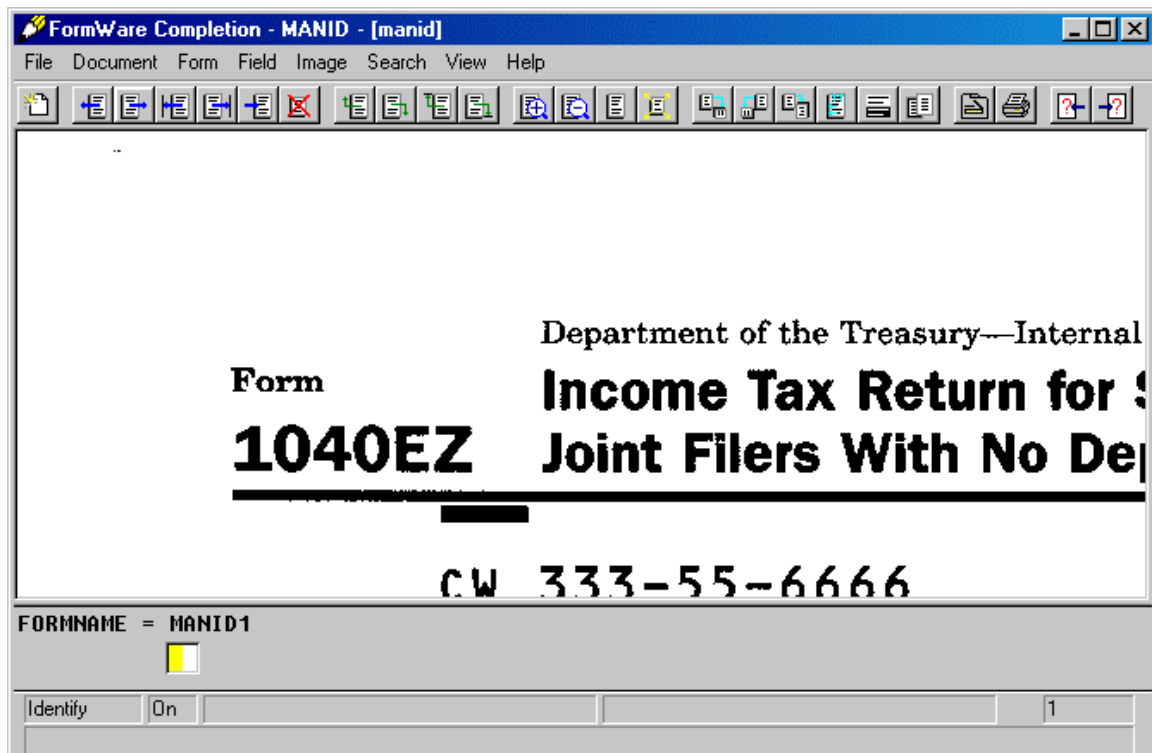
Display Option Two

Display Option Two uses the Identify Images mode of Completion and displays a Manual Form ID menu (IDM file) and Form ID field next to the image. The operator can select the menu option that corresponds to the current image. The Form ID number then displays for the selected menu option in the field at the top of the menu. The operator can also press [F5]-[F9] to display the Change Form dialog (IDM file), where he can either type a valid Form ID number or select the appropriate form name in the list. See the Configuring Manual ID to Use Display Option Two task for instructions.
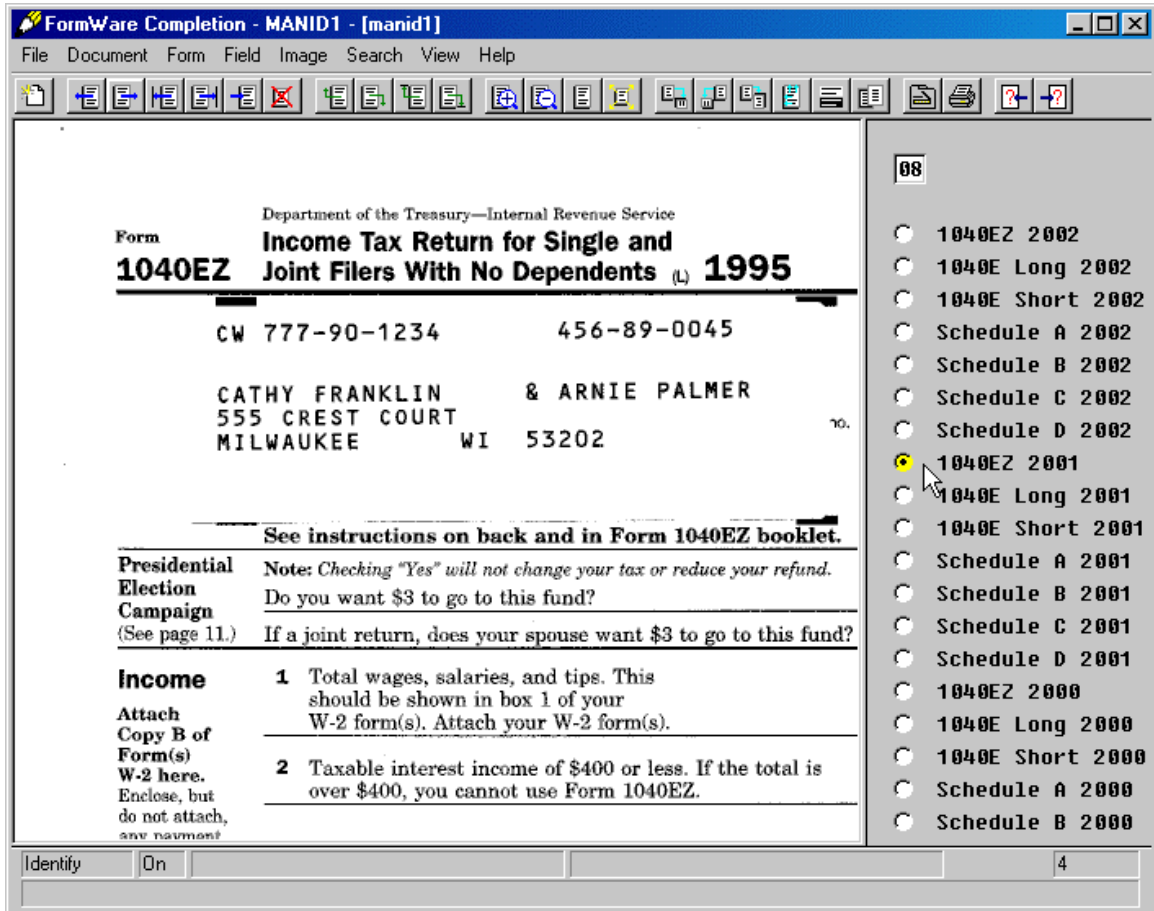
Display Option Three

Display Option Three uses the Image Quality Check/Rescan mode of Completion and displays the Change Form dialog (IDM file) over a group of thumbnail images. The operator can press [F5]-[F9] to display the Assign Form dialog, type a valid ID number in the Form Number field, and then select OK. This identifies the currently active image thumbnail. See the Configuring Manual ID to Use Display Option Three task for instructions.
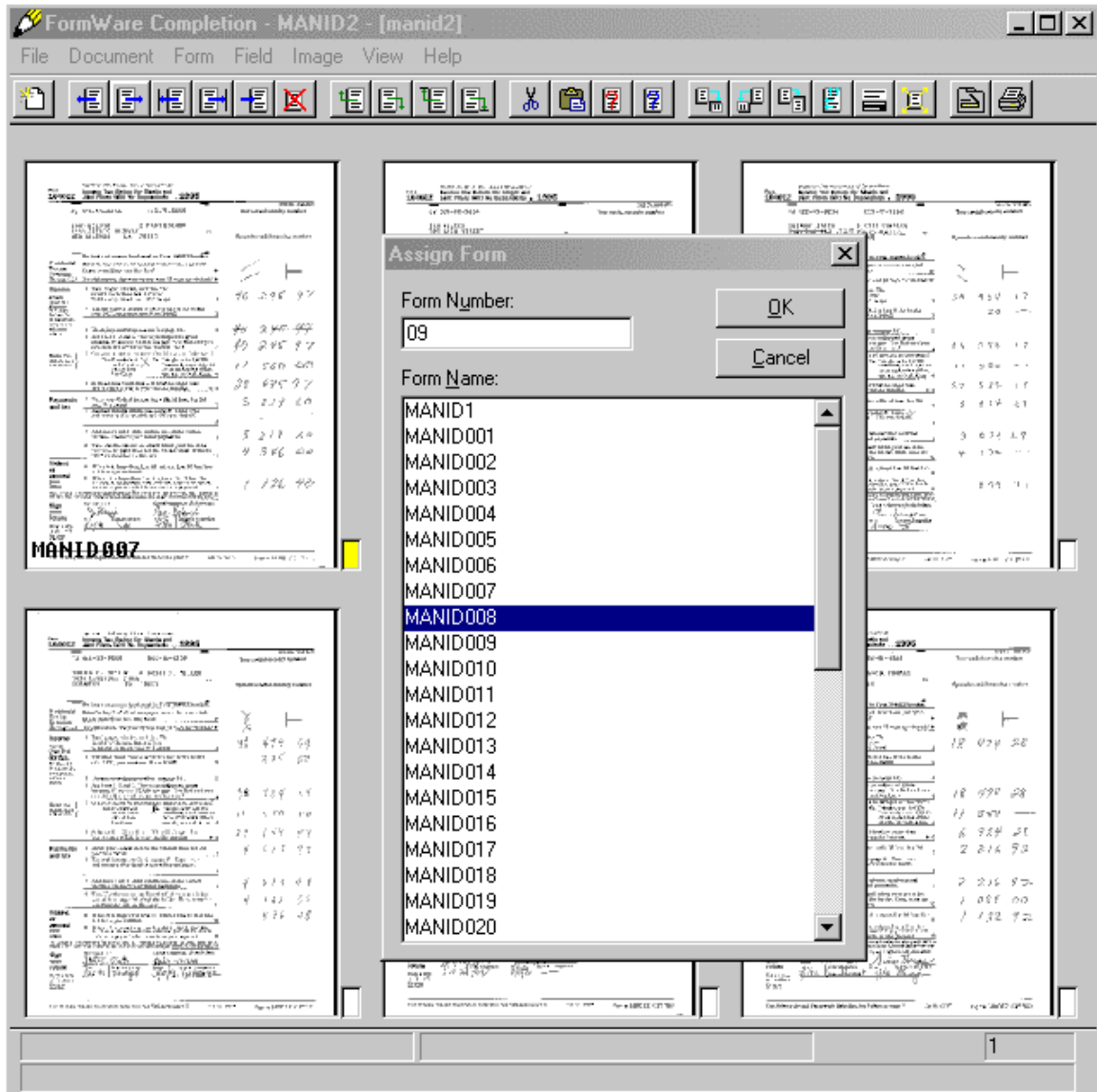
Display Option Four

Display Option Four uses the Image Quality Check/Rescan mode of Completion and displays the Change Form dialog (IDM file) with a folder structure over both the current image and a group of thumbnail images. The operator can press [F5]-[F9] to display the Assign Form dialog, and then either type a valid ID number in the Form Number field or double-click on the folder structure to select the appropriate form, and then select OK. See the Configuring Manual ID to Use Display Option Four task for instructions.
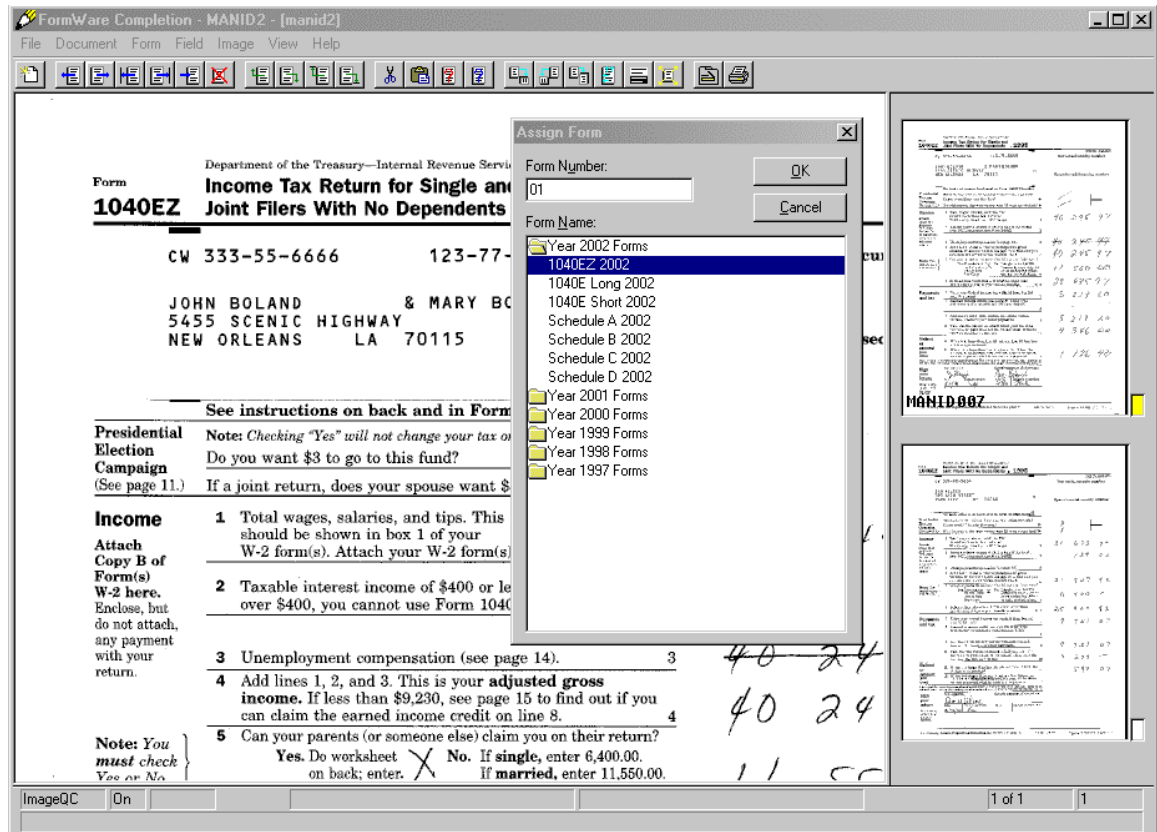
Display Option Five

Display Option Five uses the Image Quality Check/Rescan (Manual IQC) mode of Completion and forces display of the Assign Form dialog (IDM file) with collapsible folders over both the current image and a thumbnail image. *This is the preferred option when using the Manual IQC for high speed Manual Form ID.* By default, the Assign Form dialog is forced to display over the image thumbnails in the Completion window. The operator can either type a valid ID number in the Form Number field or double-click on the folder structure to select the appropriate form, and then select OK. Press [F5]-[F9] to redisplay the Assign Form dialog after closing it. See the Configuring Manual ID to Use Display Option Five task for instructions.
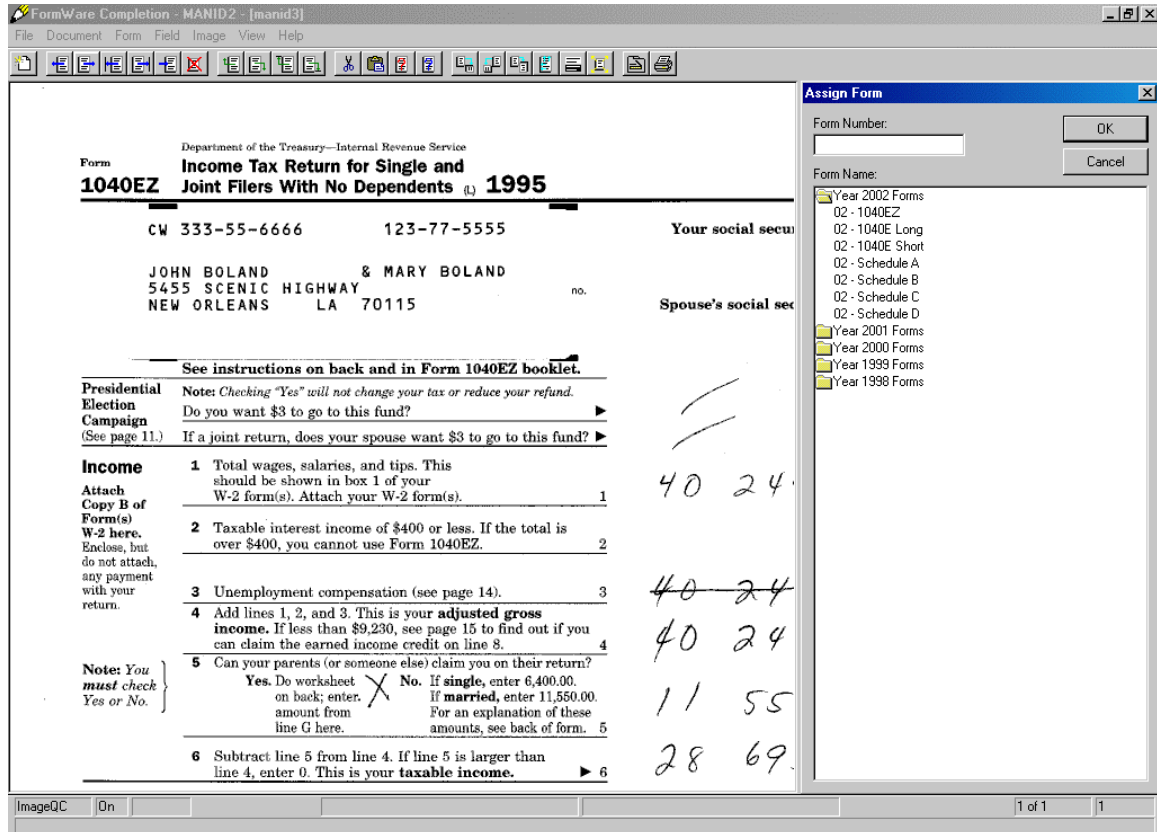
Display Option Six

Display Option Six uses any Completion mode and displays the Change Form dialog (IDM file) with collapsible folders over the current image. The operator can press [F5]-[F9] to display the Change Form dialog and either type a valid ID number in the Form Number field or double-click on the folder structure to select the appropriate form, and then select OK. See the Configuring Manual ID to Use Display Option Six task for instructions. The following illustrates the Change Form dialog used in Completion – New mode.
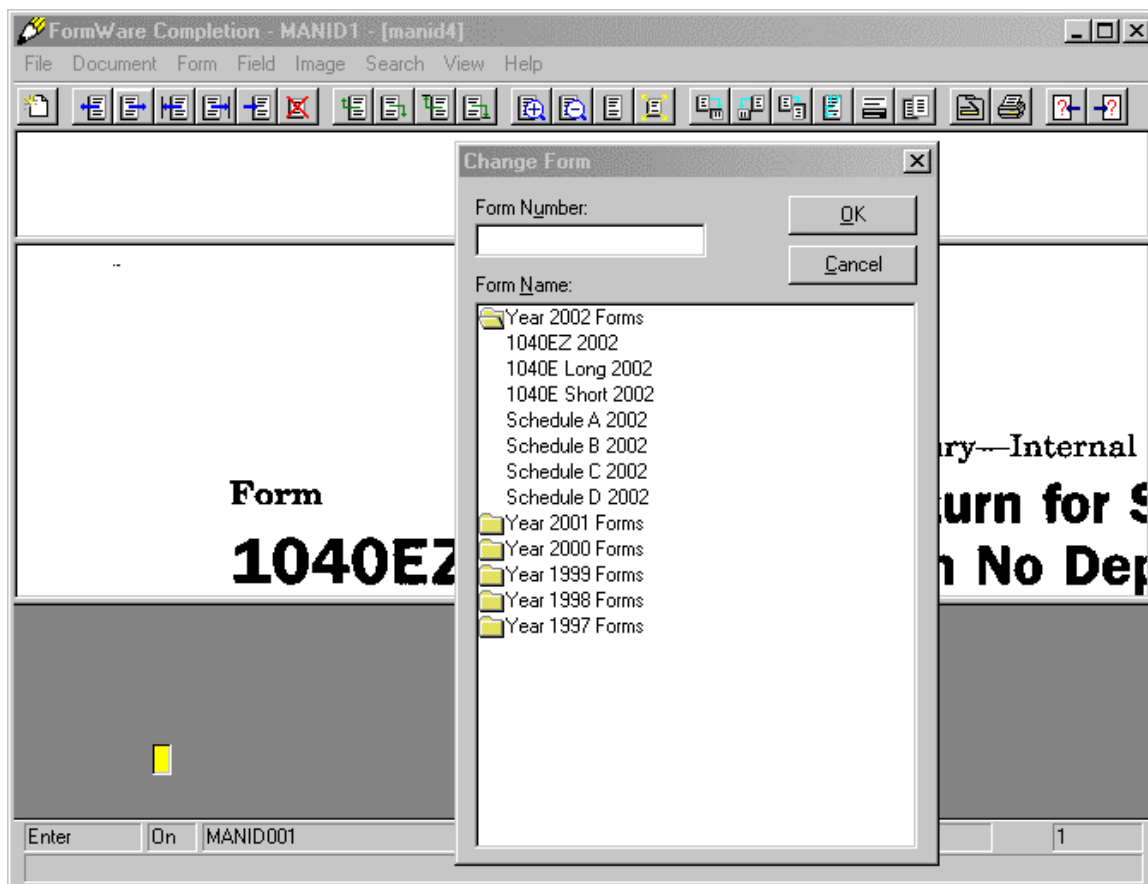
Configuring Manual ID to Use Display Option One

Display Option One uses the basic Completion – Identify Images mode (Manual ID) without any additional settings to split the screen or use a custom Manual ID menu. Press [F5]-[F9] in Completion to display the Change Form dialog (IDM file), where you can either type a valid Form ID number or select the appropriate form name in the list. Follow these steps to configure the Manual ID process to use this option:

1. Create a job with multiple forms in Designer.

2. Place at least one, two-character Any Type field on each form in the job.

3. Save your work.

4. Access **Job Properties > General > Manual ID Menu** and edit the IDM file to the appropriate specifications.

5. Modify each form number:form name line in the IDM file as you want it to display on the custom menu. See the Modifying the IDM File for a Job task for further instructions.

6. Save and exit the IDM file.

7. Create a test BDF file with the appropriate job name and a group of images appropriate for this job. Save the BDF file with the same name as the job in the \FormWare\Files folder.

8. Test the Completion process from a command line. Type:

   ```
   complete.exe –y c:\fwshare\formware\files\<bdfname>.bdf
   ```

   This runs the Identify Images mode and uses the IDM file for the job.

9. Verify that your test was successful. Retest until it is.

10. Open JobFlow Designer and create a JobFlow that contains a Completion – Identify Images task.

    a. Select the **Completion** button on the JobFlow Task toolbar.

    b. On the Completion Modes dialog, select **Identify Images** as the Processing Mode and then **OK**.

11. Save your work.

Configuring Manual ID to Use Display Option Two

Display Option Two uses the basic Completion – Identify Images mode (Manual ID) and adds additional functionality to split the Completion window and display a custom Manual ID menu next to the image. You can also press [F5]-[F9] in Completion to display the Change Form dialog (IDM file), where you can either type a valid Form ID number or select the appropriate form name in the list. Follow these steps to configure the Manual ID process to use this option:

1. Create a job with multiple forms in Designer.

2. Place at least one, two-character Any Type field on each form in the job.

3. Save your work.

4. Access **Job Properties > General > Manual ID Menu** and edit the IDM file to the appropriate specifications.

5. Add the following parameter to the very top of the IDM file:

    ```
    VWR=520,0,50,0,1,0
    ```

    This parameter overrides the Attachment Viewer properties that are typically set on the **Job Properties > Image** tab. This splits the Completion screen in two parts, one containing the image and the other containing the custom menu.

6. Modify each *form number:form name* line in the IDM file as you want it to display on the custom menu. See the Modifying the IDM File for a Job task for further instructions.

7. Save and exit the IDM file.

8. From Administration, display the \FormWare\Files folder and create a test BDF file with the appropriate job name and a group of images appropriate for this job. Save the BDF file with the same name as the job.

9. Test the Completion process from a command line. Type:

    ```
    complete.exe –y c:\fwshare\formware\files\<bdfname>.bdf
    ```

    This runs the Identify Images mode and uses the IDM file for the job.

10. Verify that your test was successful. Retest until it is.

11. Open JobFlow Designer and create a JobFlow that contains a Completion – Identify Images task.

    a. Select the **Completion** button on the JobFlow Task toolbar.

    b. On the Completion Modes dialog, select **Identify Images** as the Processing Mode and then **OK**.

12. Save your work.

Configuring Manual ID to Use Display Option Three

Display Option Three uses the basic Completion – Image Quality Check/Rescan mode (Manual IQC) without specifying an IDM file. Images are presented as thumbnails and the Completion operator can perform basic Manual IQC tasks on each image such as rotating, rescanning, or rejecting the image. Press [F5]-[F9] to display the Assign Form dialog where you can assign a form type to each thumbnail image. Once you assign a form type, the form name displays in the lower left corner of the thumbnail image. Follow these steps to configure the Manual ID process to use this option:

1. Create a job with multiple forms in Designer.

2. Place at least one, two-character Any Type field on each form in the job.

3. Save your work.

4. From Administration, display the \FormWare\Files folder and create a test BDF file with the appropriate job name and a group of images appropriate for this job. Save the BDF file with the same name as the job.

5. Test the Completion process from a command line. Type:

    ```
    complete.exe –s c:\fwshare\formware\files\<bdfname>.bdf
    ```

    This runs the Image Quality Check/Rescan mode. Press [F5]-[F9] to display the Assign Form dialog where you can assign a form type to each thumbnail image.

6. Verify that your test was successful. Retest until it is.

7. Open JobFlow Designer and create a JobFlow that contains a Completion – Image Quality Check/Rescan task.

    a. Select the **Image QC** button on the JobFlow Task toolbar.

    b. On the Image QC Types dialog, select the **Manual Image QC/Rescan** option and then **OK**.

Note: If your JobFlow contains multiple Completion tasks, the IQC/Rescan task should be the first Completion task in the JobFlow.

8. Save your work.

Configuring Manual ID to Use Display Option Four

Display Option Four uses the Completion – Image Quality Check/Rescan mode (Manual IQC), presents images in a split screen – the left side containing a full size view of the current image and the right side containing image thumbnails. The –IDM=<IDM filename>.idm parameter is used to specify an IDM file to display during the session. The IDM file must contain the VWR= parameter to define the full size image viewport in Completion. The Completion operator can perform basic Manual IQC tasks on each image such as rotating, rescanning, or rejecting the image. Press [F5]-[F9] to display the Assign Form dialog where you can assign a form type to each thumbnail image. Once you assign a form type, the form name displays in the lower left corner of the thumbnail image.

Note: In high-speed Manual Form ID mode, Manual IQC ignores the enabled Only Process Rejects option.

This option supports an expanded IDM file format, which enables you to create a hierarchical folder structure within your IDM file. See the Modifying the IDM File to Use a Folder Structure task for more information.

Follow these steps to configure the Manual ID process to use this option:

1. Create a job with multiple forms in Designer.

2. Place at least one, two-character Any Type field on each form in the job.

3. Save your work.

4. Access **Job Properties > General > Manual ID Menu** and edit the IDM file to the appropriate specifications.

5. Add the following parameter to the top of the IDM file.

   ```
   VWR=700,0,50,0,1,0
   ```

   This parameter overrides the Attachment View properties that are typically set on the **Job Properties > Image** tab. This will split the Completion screen in two parts, one containing a full size image and the other containing image thumbnails.

6. Modify each *form number:form name* line in the IDM file as you want it to display on the custom menu. See the Modifying the IDM File for a Job task for further instructions.

7. Add a folder structure to the IDM file. See the Modifying the IDM File to Use a Folder Structure task for instructions.

8. Save and exit the IDM file.

9. From Administration, display the \FormWare\Files folder and create a test BDF file with the appropriate job name and a group of images appropriate for this job. Save the BDF file with the same name as the job.

10. Test the Completion process from a command line. Type:

    ```
    complete.exe –s –IDM=<IDM filename>.idm
    c:\fwshare\formware\files\<bdfname>.bdf
    ```

    This runs the Image Quality Check/Rescan mode and displays a split screen containing both a full size view of the image and image thumbnails. The –IDM= parameter indicates the IDM file within the FormWare\Jobs folder to use during this Completion task. Press [F5]-[F9] within Completion to display the Assign Form dialog where you can assign a form type to each thumbnail image.

Note: When using this display option with Manual IQC, it is important to remember that the Manual IQC toolbars and menu options refer *only to the image thumbnails*. The full screen image is in the attachment viewer and can be manipulated by using the embedded keyboard functions of the attachment viewer. While the image view in the attachment changes each time you select a thumbnail image, the two images are not directly linked. This means that if you rotate the full screen image in the attachment viewer it does not automatically rotate the thumbnail, nor does it save the rotated image on disk. If you actually want to change the image, you have to change the thumbnail view to a full screen view and manipulate the image just as you did in previous versions of Manual IQC. Manual IQC will then prompt you to save your changes and this will rewrite the image on the disk.

---

11. Verify that your test was successful. Retest until it is.

12. Open JobFlow Designer and create a JobFlow that contains a Completion – Image Quality Check/Rescan task.

    a. Select the **Image QC** button on the JobFlow Task toolbar.

  b. On the Image QC Types dialog, select the **Manual Image QC/Rescan** option and then
      OK.

Note: If your JobFlow contains multiple Completion tasks, the IQC/Rescan task should be the
first Completion task in the JobFlow.

---

13. Double-click on the Manual IQC/Rescan task in the JobFlow Browser to display the Task
    Properties dialog.

14. In the **Client Parameters** field, type:

    ```
    -s -IDM=<IDM filename>.idm
    ```

    Where <IDM filename> is the name of the IDM file in the \FormWare\Jobs folder to use for
    this task.

Note: In the previous Manual IQC Properties dialog, -s is the code for starting the Completion
module in Image QC/Rescan mode, and -MID=Y puts Manual IQC into high speed manual Form
ID mode. The -MID=Y switch is the only parameter you need to add, and all modes of Completion
will use this file.

---

15. Close the Task Properties dialog and save your work in JobFlow Designer.

Configuring Manual ID to Use Display Option Five

Display Option Five uses the Completion – Image Quality Check/Rescan mode (Manual IQC),
presents images in a split screen – the left side containing a full size view of the current image
and the right side containing image thumbnails. The –IDM=<IDM filename>.idm parameter is
used to specify an IDM file to display during the session. The IDM file must contain two special
parameters: the VWR= parameter to define the full size image viewport in Completion and the
FWN= parameter to designate where over the Completion window the IDM menu will display.
The –MID=Y parameter is also used to automate the display of the IDM file during the session.
This is beneficial when you are using this Manual IQC task for high-speed Manual Form ID. The
Completion operator can perform basic Manual IQC tasks on each image such as rotating,
rescanning, or rejecting the image. Press [F5]-[F9] to display the Assign Form dialog where you
can assign a form type to each thumbnail image. Once you assign a form type, the form name
displays in the lower left corner of the thumbnail image.

This option supports an expanded IDM file format, which enables you to create a hierarchical
folder structure within your IDM file. See the Modifying the IDM File to Use a Folder Structure
task for more information.

Follow these steps to configure the Manual ID process to use this option:

1. Create a job with multiple forms in Designer.

2. Place at least one, two-character Any Type field on each form in the job.

3. Save your work.

4. Access **Job Properties > General > Manual ID Menu** and edit the IDM file to the
    appropriate specifications.

5. Add the following parameters to the top of the IDM file:

    ```
    VWR=700,0,50,0,1,0

    FWN=-1,0,0,0
    ```

The VWR parameter overrides the Attachment Viewer properties that are typically set on the **Job Properties > Image** tab. This will split the Completion screen in two parts, one containing a full size image and the other containing image thumbnails.

The FWN parameter designates the size and placement of the Assign Form dialog. The default location is to cover the image thumbnail portion of the Completion window.

6. Modify each *form number:form name* line in the IDM file as you want it to display on the Assign Form dialog. See the Modifying the IDM File for a Job task for further instructions.

7. Add a folder structure to the IDM file. See the Modifying the IDM File to Use a Folder Structure task for instructions.

8. Save and exit the IDM file.

9. From Administration, display the \FormWare\Files folder and create a test BDF file with the appropriate job name and a group of images appropriate for this job. Save the BDF file with the same name as the job.

10. Test the Completion process from a command line. Type:

```
complete.exe –s –MID=Y –IDM=<IDM filename>.idm
c:\fwshare\formware\files\<bdfname>.bdf
```

This runs the Image Quality Check/Rescan mode and displays a split screen containing both a full size view of the image and image thumbnails. The –MID=Y parameter automates the display of the IDM file during the session, which is useful for high-speed Manual Form ID. The –IDM= parameter indicates the IDM file within the FormWare\Jobs folder to use during this Completion task. Press [F5]-[F9] within Completion to display the Assign Form dialog if you close it.

11. Verify that your test was successful. Retest until it is.

12. Open JobFlow Designer and create a JobFlow that contains a Completion – Image Quality Check/Rescan task.

   a. Select the **Image QC** button on the JobFlow Task toolbar.

   b. On the Image QC Types dialog, select the **Manual Image QC/Rescan** option and then **OK**.

Note: If your JobFlow contains multiple Completion tasks, the IQC/Rescan task should be the first Completion task in the JobFlow.

---

13. Double-click on the Manual IQC/Rescan task in the JobFlow Browser to display the Task Properties dialog.

14. In the **Client Parameters** field, type:

```
–s –MID=Y –IDM=<IDM filename>.idm
```

Where <IDM filename> is the name of the IDM file in the \FormWare\Jobs folder to use for this task.

15. Close the Task Properties dialog and save your work in JobFlow Designer.

Configuring Manual ID to Use Display Option Six

Display Option Six enables you to display the Manual Form ID menu (IDM file) within any other Completion mode by supplying the –IDM= parameter on the command line or as a client parameter value. Press [F5]-[F9] within the Completion session to display the Change Form

dialog where you can assign a form type to each image. This option supports the expanded IDM file format, which enables you to create a hierarchical folder structure within your IDM file. See the Modifying the IDM File to Use a Folder Structure task for more information.

Follow these steps to configure the Manual ID process to use this option:

1. Create a job with multiple forms in Designer.

2. Place at least one, two-character Any Type field on each form in the job.

3. Save your work.

4. Access **Job Properties > General > Manual ID Menu** and edit the IDM file to the appropriate specifications.

5. Modify each *form number:form name* line in the IDM file as you want it to display on the Change Form dialog. See the Modifying the IDM File for a Job task for further instructions.

6. Add a folder structure to the IDM file. See the Modifying the IDM File to Use a Folder Structure task for instructions.

7. Save and exit the IDM file.

8. From Administration, display the \FormWare\Files folder and create a test BDF file with the appropriate job name and a group of images appropriate for this job. Save the BDF file with the same name as the job.

9. Test the Completion process from a command line. Use the appropriate switch for the Completion mode you want to use. Valid Completion mode switch values are:

> –e Completion – New
>
> –r Completion – Append
>
> –v Completion – Verify
>
> –c Completion – Correct
>
> –u Completion – Update
>
> –f   Completion – Remove Flags
>
> –x Completion – Read Only
>
> –q Completion – Quality Check
>
> –w Completion – Add New Images
>
> –y Completion – Identify Images (Manual Form ID)
>
> –b Completion – Batch Build
>
> –s Image Quality Check/Rescan (Manual IQC)

For example, to enable display of the Manual Form ID menu within Completion – New mode type:

```
complete.exe –e –IDM=<IDM filename>.idm
c:\fwshare\formware\files\<bdfname>.bdf
```

10. This runs the Completion – New mode with the IDM file you specify. Press [F5]-[F9] within the Completion session to display the Change Form dialog.

11. Verify that your test was successful. Retest until it is.

12. Open JobFlow Designer and create a JobFlow that contains a Completion – New task.

     a.   Select the **Completion** button on the JobFlow Task toolbar.

     b.   On the Completion Modes dialog, select **Identify Images** as the Processing Mode and then **OK**.

13. Double-click on the Completion – New task in the JobFlow Browser to display the Task Properties dialog.

14. In the **Client Parameters** field, type:

```
-e -IDM=<IDM filename>.idm
```

Where <IDM filename> is the name of the IDM file in the \FormWare\Jobs folder to use for this task.

15. Close the Task Properties dialog and save your work in JobFlow Designer.

Setting Image Window Properties for Completion

One of the ways to perform Manual ID within Completion is to display the Attachment Viewer image window during the Completion session. You can set display properties for the Attachment Viewer at the job level within Designer.

Note: You can also override these properties using the VWR= parameter if you are using an IDM file in your Manual ID process. See the Modifying an IDM File for a Job task for more information on the VWR= parameter.

To define the settings of the Attachment Viewer image window that displays in Completion:

1. Open the job in Designer.

2. Display the **Job Properties > Image** tab.

3. In the **Attachment Viewer** area, set options that are used for both Identify Images mode (Manual ID) and the optional Attachment Viewer component.

4. Select the **Display** check box to display the image window during Completion.

5. Select the **Include current image** check box to include the current image in the image window.

6. Select the **Scale to gray** check box to apply the scale to gray properties to the image in the image window.

7. In the **Width** field, select the horizontal pixel count the image window will use to display the image. (The **Height** field value is not applicable for the Identify Images mode.)

8. In the **Scroll** field, select the pixel count the image will scroll when the operator uses the up and down arrow keys in the image window.

9. In the **Scale** field, set the scale factor to be the default in the image viewer.

10. In the **Rotate** field, set the rotation factor to be the default in the image viewer.

11. Select the **Save All** on the Designer window toolbar to save your settings.

Writing Rules for Form ID Fields

To further define your Form ID process, you can assign more than one Form ID field to a data form. Form ID fields can act independently or their results can be associated with rules that will tie the fields together.

If you do not associate a rule to the different Form ID fields, the Form ID component bases its processing on data form linkage, then on the Form ID field number until it finds a possible match. This means that the Form ID component first goes to the data form with the lowest link value and processes the Form ID field with the lowest field number in that data form. If it finds a match, the image is identified as being of that form type and then the Form ID component proceeds to the next image. If the first Form ID field is not a match, then the Form ID component processes the next Form ID field on the data form. The Form ID component cycles once through all the Form ID fields on the data form. If all Form ID fields are rejected, it proceeds to the next data form to do the same. No image in the batch is given a form number of 99 until all data forms containing Form ID fields are processed.

Note: If you have defined more than one Feature Form ID field on your data form and have not assigned a rule to any of those Form ID fields, you must note the order in which you set up the data forms and Form ID fields within the job. This impacts the processing of Form ID.

You can also use rules to associate the Form ID fields on a data form. Specifying a rule allows you to use complicated decision logic to associate results from multiple Form ID fields on a data form to determine form identification. Before you can specify a rule for the Form ID field, you must specify a rule on the form level, via the Form Properties, Edits tab. Once you have selected a rule file for the data form, select the **Browse** button adjacent to the **Rule Name** field on the Field Properties dialog, ID tab to display a list of the currently available rules. You can then select the **Rules** button on this tab to launch an editor that enables you to edit the rule. If you have not specified a rule, you can edit the entire rule file.

You can create any number of field associations using rules. For example, you can create a rule that requires all Form ID fields on the data form to match for the image to be identified as the specified form type. You can also create a rule that returns successful form identification when three out of five of the Form ID fields match on the data form.

For more information on rules, refer to the Rules section of the FormWare Programming Manual.

Enhancing an Image

Image Enhancement provides further improvement of pre-processed images to prepare them for Recognition. Enhancement is the key to obtaining good OCR results. You can run the Image Enhancement process after IPP are performed, or in lieu of IPP. Unlike IPP, which is performed on an entire image, Image Enhancement performs functions on the portion of the image defined within a recognition zone. This enables you to enhance an image at a more detailed level than the page level.

The result of the Image Enhancement process is an enhanced image that can either overwrite the original image or (by default) be output to an alternate directory. This directory has the same name as the input directory but has a .enh extension. For example, if your input images are in the \FormWare\Images directory, the enhanced images are output to the \FormWare\Images.enh directory. If you want to specify an alternate directory as the default, you must set the ENHPATH=<path name> parameter in the BDF with the path to the existing alternate directory.

You can use the Batch File Initialization Data option in the Scan module to place this parameter in the BDF during a scanning or importing process.

Image Enhancement includes:

- **Registration** – Deskews and scales images to match the setup image.
- **Zone Removal** – Removes lines and text that surround a recognition zone, and can also apply image filters for removing image noise and smoothing dot matrix print.
- **Form Removal** – Combines the image pre-processing (IPP) functions of cropping, deskew, registration, and line removal with a template-based "background removal" technology.

Registering an Image

Registration zones instruct the recognition engine where to look for data on each image and make adjustments for production image variations. When registration zones exist, the recognition engine checks the X and Y coordinates of these registration zones on each production image and shifts the image so that all the data is in the same place as the setup image.

A data form is registered by defining specific markers, called anchors, on form background text. You need to specify at least one text, or two or more non-text registration anchors for each data form type used. With one text anchor, or two non-text anchors, the registration process can correct for X and Y translation and image skew . With three anchors, registration can also optionally correct for image scale (magnification, or linear image stretch or squeeze).

Form registration consists of three components:

- A **registration field** on the data form.
- A **registration zone** on the image.
- A **registration anchor** within the registration zone.

Working with Registration

When defining a registration field and corresponding registration zone, avoid zoning a registration mark close to other, similar marks or to any text you want to capture. If you zone a registration mark on the setup image too close to data that you want to capture, you run the risk of inadvertently writing over the registration anchor zone. In addition, when you define more than one registration anchor, place the anchors as far apart on the data form as possible. This gives you better registration results. As a rule, you should not define more than three or four anchors on a single data form.

If you enable scale correction, a search on the production image for anchors in the order they were defined, finds the most widely separated three anchors and uses them as registration points. If you disable scale correction, the first two anchors found are used. If only one anchor is found, and that anchor is a text anchor, registration proceeds. If no text anchor or one non-text anchor is found, the image fails registration with a form-level error, and no recognition is performed on the image. If scale correction is enabled, at least three anchors must be found regardless of anchor type—even if one or more of the anchors is a text registration anchor—or the image fails registration.

It is essential that you test registration anchors on as many images from your suite of test images as possible before you define recognition zones on a data form. There are two possible failure modes for registration:

- *Failure to Register* – This is caused by an inability to locate sufficient registration anchors on an input image. You can sometimes remedy this situation by reducing the required confidence for the anchor points, or by defining additional registration zones. An anchor matching the wrong feature on an input image causes false registration. This typically occurs when you define an anchor too close to another area of the image that is similar in appearance. You must place anchors far enough from an ambiguous mark so that the expected scanner-induced image shift does not shift that mark into the search area for the anchor.

- *False Registration* - False registration can also be caused by form variants. Often, a user may initially believe that there is only one printed appearance of a form. However, on closer examination, form images can have significant variations caused by print layout— an anchor point on the gold image may be located in a different place on another variant. In this case, you should select a different anchor point that is in common across all forms or use Form ID to treat the form types as different forms.

Often, registration properties must be adjusted to cause all images in the test suite to register properly.

Creating Registration Fields and Zones

Use the Registration field type to assign registration anchors to the setup image. These anchors are used to determine the location of the data fields for the production images. The registration process takes place during Enhancement.

The registration anchors are applied in the order you create them, unless you change the search order for one or more anchors. You can define any number of registration anchors on a single data form. However, you must define at least one text anchor or two non-text anchors on the data form to enable the registration process.

Note: Before you define registration anchors, you must have a gold image. A gold image is a nearly perfect image. One that contains very few speckles, or noise, and is not skewed. If you are using IPP, you should run IPP functions against the image before you define registration anchors.

To define a registration field and a corresponding zone on the setup image:

1. Open the job and form you want to register in Designer.

2. Select the **Registration** button on the Field Types toolbar and then select a place on the data form. A one-character, blue field is added to form.

3. Adjust the display of the setup image in the Image Viewport so the feature you want to use as a registration anchor feature displays.

4. With the registration field selected, draw a recognition zone around the registration anchor feature on the setup image. The item you zone on the image for registration should be a mark that is common to all images of this type (e.g., logo, intersection, text). Define registration zones as far apart as possible, but at least ½ inch from the edge of the image, so that if the edge of an image is cropped too closely or torn, the registration anchor is not lost.

5. Select the **Save All** button on the Designer window toolbar to save your changes.

Now that you have defined one form registration field and zone on the setup image, proceed to the next task where you associate that field and zone with a registration anchor.

Creating Registration Anchors

By default, when performing registration on an image, the anchors are applied in the order in which they were created. You can define any number of registration anchors on a single data form. However, you must define at least one text anchor or two non-text registration anchors on the data form to enable the registration process. You should also define registration zones as far apart as possible. At least ½ inch from the edge of the image, so that if the edge of an image is cropped too closely or torn, the registration anchor is not lost. If stretch correction is enabled, then you must define at least three anchors, regardless of anchor type, on the data form.

After you have created a registration zone for a registration field on the setup image, you can define registration anchors. To create a registration anchor:

1. Double-click on the registration field to display the Field Properties, Registration tab .

2. Select the **Registration** button. The Registration Anchor dialog displays. The following information displays on the status bar at the bottom of this dialog.
    - o X - The X coordinate of the current cursor position.
    - o Y -The Y coordinate of the current cursor position.
    - o Anchor Position - The location of the anchor point on the image (X and Y coordinates).
    - o Confidence - The confidence returned after testing the anchor properties, represented as a percentage of accuracy.

3. Select a button on the Registration Anchor toolbar that most closely matches the item on the image you have zoned. If you select an anchor toolbar button that does not create the anchor zone correctly, choose another anchor tool. Note the following:
    - o The "T" line anchor tools may not work as accurately as the "L" or "+" line anchor tools.
    - o The "L" tools operate differently than the "T" or "+".
    - o The "T" and "+" tools create the best match within a search area, and must be unique.
    - o The "L" tools will match the outermost corner within the search area. For example, if an image contains a grid of horizontal and vertical lines in the search area. Then use an "L" tool to match the "L" shape in the lower left corner, or an inverted "L" to match the "L" shape in the upper-right corner.
    - o Text anchors can be very reliable if the text line is over two inches long and unique in its properties (e.g., text size).

4. Move your cursor over the registration item and click once. A blue box is drawn around the item.

5. Stretch the blue box to expand the search area if necessary. This allows for greater image shifts, as long as there are no ambiguous image features nearby. This zone is used when the recognition engine performs registration. The anchor must appear within this zone on the unregistered image to be located.

6. Use the following toolbar buttons to adjust the image view area:
    - o **Zoom In** – Magnifies the image appearing in the image view area.
    - o **Zoom Out** – Decreases magnification of the image.
    - o **Home** – Returns the image display to the original image view area that displayed when you first accessed the Registration Anchor dialog. This enables you to return to a starting point if you have scrolled or panned to another location on the image.
    - o **Select** – Activate, move, or resize the anchor zone.

7. Select the anchor type to match area you are trying to register against. Anchor types include:

    o **Line Anchors -** Defines a registration anchor of the specified shape within the registration zone. Select the appropriate line anchor that best illustrates the feature on the image you want to use as a center point for the registration anchor.

    o **Shape Anchors** - Defines a registration anchor of the specified shape within the registration zone. Select the appropriate shape anchor that best illustrates the feature on the image you want to use as a center point for the registration anchor.

    o **Text Anchor** - Defines a registration anchor using text on the image. When selected, use the crosshair pointer to zone the text on the image you want to use for a registration anchor.

8. Select the **Test Anchor** button to visually review where an anchor is detected in the current image, and display the confidence rating in the **Confidence** field of the status bar. If the anchor is found, a small red box shows the location of the anchor point. Adjust the anchor until it registers correctly.

9. Select the **Edit Anchor** button to modify an anchor or its properties after a registration test. This function also reverses the deletion of an anchor. If you previously deleted an anchor, it restores the original anchor and enables you to modify the anchor properties.

10. Select the **Set New Anchor** button to change the location of the "hot spot" of an anchor from its original location. The anchor location (blue box) will be set from the current detected anchor (red box). Typically, you should only use this function when you want to change the image without having to re-train the anchor. In this case, you need to adjust the hot spot to correspond to the new anchor location on the new image after performing a test.

11. To remove an anchor, select the **Delete** button. You cannot create a new anchor zone until you delete the existing zone.

12. Select **OK** to return to the Field Properties dialog.

13. Select **Save All** on the Designer window toolbar to save your registration anchor.

Modifying Anchor Properties

Once you have set an anchor, you can review or modify its properties. The properties that display depend upon the type of anchor (i.e., line, shape, or text). The type displays in the Anchor Properties dialog title bar. To display the anchor properties:

1. On the Registration Anchor dialog, define a registration anchor.

2. Select the anchor on the image.

3. Select the **Properties** button. The Anchor Properties dialog displays.

4. Each anchor type displays values on the Page tab. All registration zones are ranked to determine the priority in which they are used to register an image. Valid values for these fields are from 1 to 999, with 1 being the highest priority. If you leave these setting to their default values, the anchors are applied in the order defined. To modify these values:

    a. In the **X Reliability Ranking** field, select the number that specifies the order in which the detection of this anchor should be used to determine X-translation of the unregistered image.

    b. In the **Y Reliability Ranking** field, select the number that specifies the order in which the detection of this anchor should be used to determine Y-translation of the unregistered image.

c.   In the **Skew Reliability Ranking** field, select the number that specifies the order in which the detection of this anchor should be used to determine skew of the unregistered image.

d.   Select the **Enable Image Scaling** check box if images can be stretched, squeezed, or scaled during the scanning process. If you check this option, you must define at least three registration zones on the data form. Just before registration, the recognition engine will automatically adjust the X and Y zone coordinates to compensate for scaling.

Note: To use this feature, check the **Enable Image Scaling** option for each registration anchor you define.

5.   Select the Line Anchor tab to define anchor values for the current registration anchor. This tab is very similar for the different line, shape, and text registration anchor types. See the following sections for instructions on setting the fields on this tab.

6.   Review and modify the properties and select **OK** to close the dialog.

Setting Line Anchor Properties

An anchor point defines the X- and Y-coordinate locations of the anchor on the image. In the Registration Anchor dialog, this point is displayed as a small white box in the center of the anchor mark. It is placed vertically mid-way up the character. You cannot change these coordinate values.

1.   Use the values in the Search Area to specify an area around the registration anchor that will be examined for the presence of an anchor on the production image. This area displays as a blue box around the anchor mark on the Registration Anchor dialog. These values are:

o   In the **X** field, select the X-coordinate location of the registration zone search area on the image.

o   In the **Y** field, select the Y-coordinate location of the registration zone search area on the image.

o   In the **Width** field, select the width of the registration zone search area on the image in pixels.

o   In the **Height** field, select the height of the registration zone search area on the image in pixels.

2.   In the **Line Thickness** field, select the thickness of the line used for this registration anchor in pixels. You only need to change this value if you determine that most production images are lighter or darker than the setup image, or if the thickness of the anchor lines in the production images is larger or smaller than the setup image.

3.   In the **Confidence Threshold** field, type the minimum confidence level for this registration anchor. If this confidence level is not met, the anchor will not be found. If the Image Enhancement engine fails to locate your registration anchor, you might need to lower this value. This value is a decimal percentage, with 1.0 = 100 % (e.g., .75 = 75%).

Setting Solid Anchor Properties

An anchor point defines the X- and Y-coordinate locations of the anchor on the image. In the Registration Anchor dialog, this point is displayed as a small white box in the center of the anchor mark. It is placed vertically mid-way up the character. You cannot change these coordinate values.

1. Use the values in the Search Area to specify an area around the registration anchor that will be examined for the presence of an anchor on the production image. This area displays as a blue box around the anchor mark on the Registration Anchor dialog. These values are:

    o   In the **X** field, select the X-coordinate location of the registration zone search area on the image.

    o   In the **Y** field, select the Y-coordinate location of the registration zone search area on the image.

    o   In the **Width** field, select the width of the registration zone search area on the image in pixels.

    o   In the **Height** field, select the height of the registration zone search area on the image in pixels.

2. In the **Covered Area** field, select the thickness of the shape used for this registration anchor. This value is in pixels.

3. In the **Confidence Threshold** field, type the minimum confidence level for this registration anchor. If this confidence level is not met, the anchor will not be found. If the Image Enhancement engine fails to locate your registration anchor, you may need to lower this value. This value is a decimal percentage, with 1.0 = 100% (e.g., .75 = 75%).

Setting Text Anchor Properties

For a text anchor, you should draw your initial search area tightly around the anchor text, and then stretch the zone to expand the search area. Text anchors are unique in that you can register an image using a single text anchor. Otherwise, a minimum of two line anchors is required to register the image. The line of text used should be fairly long—typically at least five inches—if you are registering the image on one text anchor only. Otherwise, the skew angle of the test image will not be computed accurately. Also, you can draw a search area around two or more sections of text that have large empty spaces between them, as long as they are on the same line. This is another way to make the text line long enough to compute the skew angle accurately.

An anchor point defines the X- and Y-coordinate locations of the anchor on the image. In the Registration Anchor dialog, this point is displayed as a small white box located vertically on the leftmost outer edge of the first character. It is placed vertically mid-way up the character. You cannot change these coordinate values.

1. Use the values in the Search Area to specify an area around the registration anchor that will be examined for the presence of an anchor on the production image. This area displays as a blue box around the anchor mark on the Registration Anchor dialog. These values are:

    o   In the **X** field, select the X-coordinate location of the registration zone search area on the image.

    o   In the **Y** field, select the Y-coordinate location of the registration zone search area on the image.

    o   In the **Width** field, select the width of the registration zone search area on the image in pixels.

    o   In the **Height** field, select the height of the registration zone search area on the image in pixels.

2. In the **Fusing Factor** field, select the numeric value that determines which dots within the text anchor area should be fused. This is used for dot matrix characters; it scales the image to lower resolution and fuses splotches on the image to form characters. A fusing factor of 1 (default) specifies that finding the text anchor should be performed at full resolution. A fusing

factor of 2 merges every other pixel prior to finding the text anchor, 3 merges every third pixel, etc. The valid range of fusing factor values is 1 through 5.

3. Select the Constraints tab. The fields on this tab are critical to the successful use of text anchors. Set these values to include the text you want as part of your anchor and exclude extraneous text. Use the X-Y cursor display in the Registration Anchor status bar to measure the height and width of characters to set these properties more precisely.

4. In the **Minimum Character Count** field in the Line Constraints area, select a filter value. This filters out any lines that do not contain at least the specified number of characters. Note that one or more characters can fuse together depending on the quality of the image and the value in the **Fusing Factor** field. If you use this field, set it to the smallest value that still distinguishes the text line from other lines.

5. In the **Minimum Line Width** field, select the distance from the left side of the first character on the horizontal text line to the left side of the last character to count. This value is in pixels.

6. Set the values in the **Character Constraints** area, to filter out any markings that are not within the given constraints. A setting of 0 (zero) disables the filter. Use the minimum settings to filter out any noise on the image and any smaller characters that lie within the registration zone. Use the maximum settings to filter out larger characters that lie within the registration zone. Use caution and set the values realistically.

    o **Character Height Min** - Select the numeric value of the minimum character height you want to include. All characters below this height are removed.

    o **Character Height Max** – Select the numeric value of the maximum character height you want to include. All characters above this height are removed.

    o **Character Width Min** – Select the numeric value of the minimum character width you want to include. Any character thinner than this width is removed.

    o **Character Width Max** – Select the numeric value of the maximum character width you want to include. Any character wider than this width is removed.

Testing Image Registration

When a registration anchor is first defined on the Registration Anchor dialog, you test that anchor using the **Test** toolbar button. After your registration anchor test is successful, you should test the registration marks against the entire data form and job.

Test both registration and removal by checking the **Image Enhancement** option on the Test page, from either the Job Properties or Form Properties dialog within the Designer module. Registration and removal are just two of a number of processes that you should test at the job, form, and field level.

The enhanced registration review tool allows you to test, review, and "auto-tune" registration settings at the form-level within Designer. Once you set up form registration fields, you can perform these enhanced registration functions using the options on the Form Properties, IPP page within Designer.

To use the features on the Form Properties, IPP page, it is important to understand the logical order of the tasks you should perform. Use the list that follows as a guideline for testing, correcting, and auto-tuning registration fields on forms prior to moving them into production.

Caution: The BDF you use for testing here cannot be used for any other testing purposes. Registration test information will write to this BDF rendering it unusable for any other testing. You

may want to copy an actual BDF from your job, rename it to designate that it is a test file, and use it for this test.

To test registration on a batch of images:

1. Display the Form Properties, IPP tab.

2. In the **Setup and Test** area, select the **Registration** option.

3. Select the **Test** button. The Browse Files dialog displays.

4. Select a BDF and then **Open**.

5. Select the **Review** button to display the Batch Test Review dialog. Use this dialog to:
    o   View batch images and their registration fields
    o   Correct registration field zones, if necessary
    o   Accept correctly registered images
    o   Remove incorrect images from the BDF, if necessary

The current image for the selected BDF is displayed in the right-hand panel, and each registration field is enclosed in a red border. In the left-hand panel, four windows contain close-up views of the registration field zones. Up to twelve registration fields can be viewed for a given form. Use the left (previous) and right (next) arrow buttons above the windows to scroll through the registration fields.
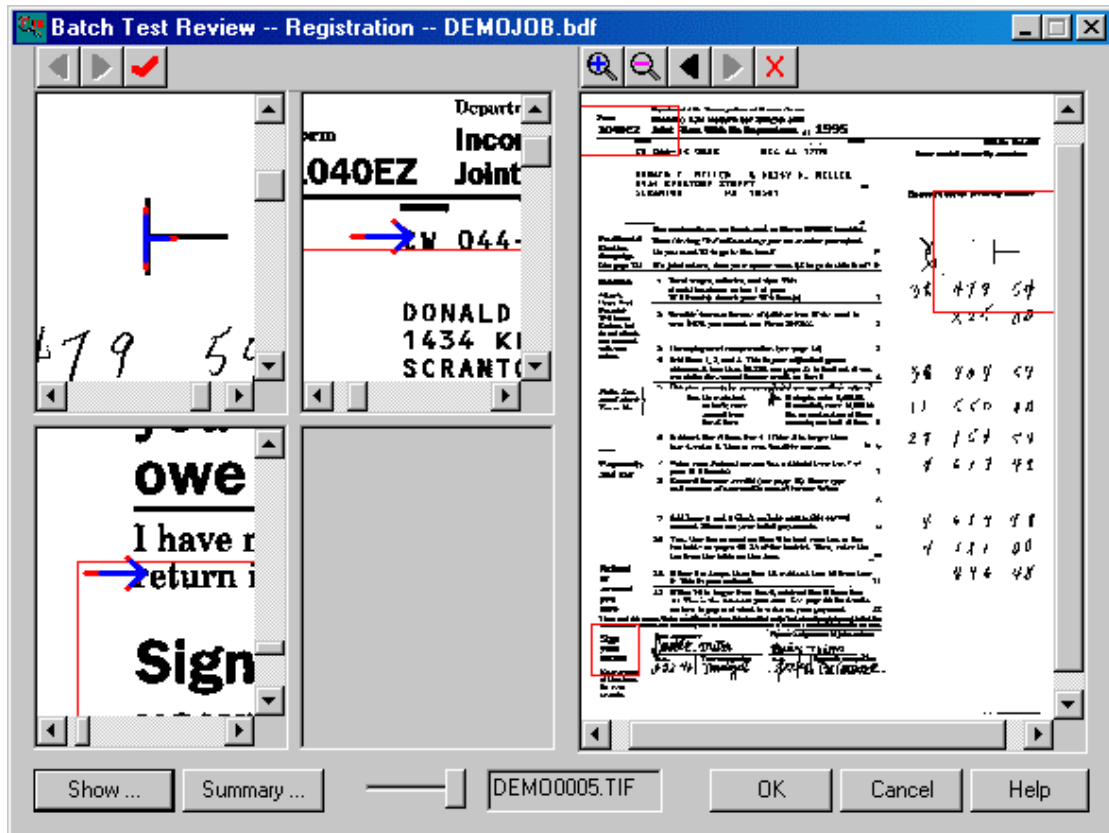
Each Registration Field Image window contains a registration field from the current image. The field registration search area is outlined in red. The red border displays in one of two ways: solid red indicates the registration field was used to register the form, a dashed red line indicates that field was not used to register the form. Additionally, each registration field for

which registration data was found contains a red overlay that indicates the *hotspot*, or center, of the data.

If you have accepted the image, with or without a correction to a registration field, a blue overlay is also present. Its location within the window depends on whether you have corrected the registration field to indicate a different data hotspot. If you made no correction, the blue overlay (inside solid line box) appears directly on top of the red overlay (inside dotted line box), but is slightly smaller in size.
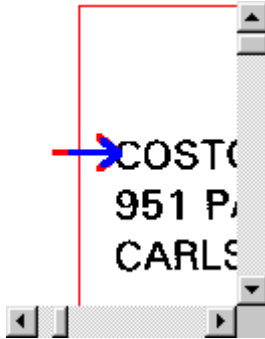
The blue overlay is moved (with a mouse click) when you are correcting a registration field and then displays elsewhere within the red zone border. If you click outside the red zone area, the blue arrow disappears.

The shape of the red and blue overlays depends on the type of registration anchor associated with the registration zone. For example, a text anchor will have red and blue arrows as shown here:
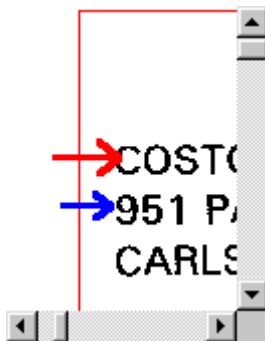
A line anchor will have red and blue square overlays as shown here:
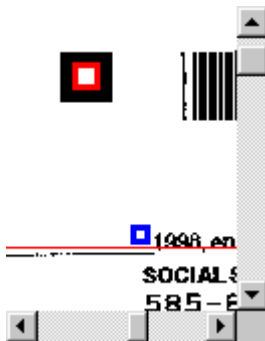
The Full Image window displays the current image with the registration zones outlined in red. Use the buttons on the toolbar above the image window to:

- o   + Zoom in on the image
- o   − Zoom out on the image

- o ▸ Advance to the next image in the batch
- o ◂ Return to the previous image in the batch
- o ✕ Remove the current image from the batch or reinstate an image that has already been removed

6. Review all registration fields for the current image. Use the previous and next buttons above the windows to scroll through the fields. Up to twelve registration fields are supported for each image. If you determine that all registration fields are correct, select the **Accept** button. This indicates that all found registrations are correct and automatically advances the display in the right-hand pane of the dialog to the next image in the BDF.

7. Occasionally, registration fields will mis-register or not register at all. To correct a mis-registered (incorrect) field, within the registration field image window, click on the correct location of the data's hotspot location in the search area box (i.e., within the red border). A blue overlay displays in the mouse click location. The blue overlay indicates the correct location in which registration data (answer) should be found.

   For any fields that need the data hotspot removed, scroll to an area outside the red border and click. This leaves no blue overlay and indicates that it is acceptable to *not* find an answer for this registration field.

   When a field is not found, typically it is due to one of two reasons:
   - o The actual registration data is located outside of the search area, preventing you from indicating the correct answer with a blue overlay. In this case, you must expand the search area to include the location of the answer.
   - o The anchor type for the field is too restrictive. For example, a field is a CROSS line anchor type, but one of the legs of the cross is obscured on some images. By changing the anchor type to an L line anchor, you enable the registration to succeed.

8. Select the **Summary** button. The Registration Test Summary dialog displays the following information. In these descriptions, the term "answer" is the registration location. The answer area, indicated by the blue overlay, results from one of two actions: user approval (i.e., you selected the **Accept** button) or you corrected a field to indicate where registration data should have been found.
   - o **Test #** - Instance of the test run. As you continue to run tests, only the six most recent, with the latest test at the top, will display their results in this dialog. The oldest result files are deleted as new tests are generated. Summaries are generated as text files in the \FormWare\Tmp\Diagnose\Enhrev a .sum extension.
   - o **Batch** - Name of the BDF or batch data file for which test data has been summarized.
   - o **Total Images** - Number of images in the batch, minus any images that were removed by the **Removal** toolbar button on the Batch Test Review dialog.
   - o **Registered Images** - Total number of images in the batch that successfully registered.
   - o **Accepted Images** - Number of registered images whose registration locations were found or corrected and which were then accepted by the **Accept** button on the Batch Test Review dialog.
   - o **Correct Images** - Number of answered images whose actual found registrations match the answers. For an image to be correct, all the registration fields that were found *and* used to register the form match their answers and the image contains no registration fields that were *not* found, but had answers indicating they should have been found. For example, a form has four registration fields with L line anchors. The system was able to register the form using any two of them. Answers were entered

for fields 1, 2, and 3. In Case 1, the system registered the image using fields 1 and 2, and also finds 3 and 4. The image is counted as correct because 1 and 2 matched answers, and 3 and 4 were not used to register the image. In Case 2, the system registered the image using fields 1 and 2 and does not find 3 or 4. The image is counted as incorrect, because 3 was not found at all, but had an answer indicating it should have been found.

- o **Incorrect Images** - Number of images whose actual found registrations did not all match the answers. These images have registration marks, but they are outside of the detection area.

9. Review the statistics for the test you generated and select **OK** to close the dialog.

10. Select **OK** to return to the IPP tab.

11. Select the **Auto Tune** button to launch the Auto Tune process, which:

- o For each unaccepted image, auto tune each failed registration field, using the Registration tool's creation tools, Auto Tune attempts to create a new candidate registration specification at the hotspot indicated by the user. Typically, this is the same registration type (e.g., CROSS, L line anchors) as the existing specification. In some cases, simpler anchors are tried if the original is a complex anchor. For example, if the original specification is a CROSS, then a CROSS and T line anchors, in all four rotations, are also tried.

- o Sort the candidate's specifications by how many correct hotspots they matched.

- o For all image candidates, test the images to determine how many of the accepted images are made incorrect by the change. Choose the best candidate that causes the fewest errors in accepted images. If the selected candidate causes *no* errors in previously accepted forms, the existing registration field specification is automatically replaced with the selected one. Otherwise, you are presented with the information, and given the option to use the selected candidate or keep the original one.

Once Auto Tune has completed, the Batch Test Review and Registration Test Summary dialogs display. Review the test statistics again.

12. Close the Registration Test summary dialog to return to the Batch Test Review dialog.

13. Select the **Show** button. The Items to Show dialog displays. The options on this dialog enable you to place filters on the images that display in the Batch Test Review dialog. The term *answer* is defined as a registration location. The answer area, indicated by the blue overlay, results from one of two actions:

- o User approval, i.e., you have accepted the image and its registration fields by selecting the red checkmark **Accept** button.

- o You corrected a field to indicate where registration data should have been found. To correct a field, right-click the correct location within the registration zone for that field.

14. In the **Forms** area:

- a. Select the **All** option to view multiple form types associated with the job.

- b. Select the **One** option to view only one form within the job.

- c. If you selected the **One** option, in the field within the Forms area, select the form you want to view. The number in parentheses identifies the form number within this job. Any images in the BDF identified as another form type are skipped during the review.

15. If you selected the **One** option within the **Forms** area, in the **Fields** area:

    a. Select the **All** option to view all registration fields on the current image.

    b. Select the **One** option to view one field.

    c. If you selected the **One** option, in the field within the Fields area, select a specific registration field to review. The number in parentheses identifies the form number associated with this field.

16. In the **Images** area, select various filters for the images in the BDF. You can select any combination, or all, of the options for the images based on the status of the registration fields you want to review. Select from the following options:

    o **Accepted but Incorrect** - Include accepted images with registration fields that registered incorrectly during the test process. These images will show registration data incorrectly located in one or more fields.

    o **Accepted and Correct** - Include accepted images with registration fields that registered correctly during the test process. These images will show registration data properly located in all fields.

    o **Ignore Incorrect unused fields** - Include images with fields that were not used to register the form during the test process. By default, unused fields that are found, and have answers but do not match their answers, count as incorrect fields. Selecting this option causes those fields not to be included in the review. This option is not turned on as a default (i.e., unless you select the option, the review will include incorrect, unused registration fields).

    o **Unaccepted** - Include images with fields that have not been accepted (i.e., no answers were found). By default, this option is turned on.

    o **Failed Registration** - Include images with fields that failed registration completely. For example, you may want to include failed registration images so that you can visually inspect them to determine whether those images: 1) Should register using the existing set of registration fields, but need additional tuning of those fields (i.e., images that show valid registration image data within the highlighted registration image zones); 2) Have a different form layout, and therefore need additional registration fields (i.e., images that show missing registration image data within the highlighted registration zones); and 3) Are flawed (e.g., field cross-out, torn image), which should be excluded from further testing by removal from the batch.

    o **Removed** - Include images that have been removed from the batch (using the red **X** button on the Batch Test Review dialog).

17. If you have not previously altered the default settings on this dialog, only images with incorrect registration fields (even if you accepted them in step 2), unaccepted images, and images with failed registration should still be displayed within the Batch Test Review

    a. For each image, correct unfound or mis-registered fields, and then accept the image.

    b. If necessary, remove images with failed registration from the batch.

18. Select other filter options to display any other images you want to review. Selecting all options will display all images within the batch again.

19. For images with failed registration that have not been removed from the batch, display the Registration Anchor tool to manually adjust the search area and reset the registration zone. To display the Registration Anchor dialog, right-click on a registration field displayed in one of the Registration Field Image windows within the Batch Test Review dialog.

Setting Zone Removal Properties

When you scan a paper form, some of the background drops out of the image. This can depend on the color of the ink on the paper form and the color of the scanner bulb. In general, black ink does not drop out while lighter shades of red, blue, or green can. A form background is the appearance of the preprinted paper form. A form background often includes lines, characters, logos, bar codes, check boxes, etc. The removal of non-data related items is often necessary to enhance recognition results.

Removing the unneeded data from a scanned document can significantly improve the recognition of the text on the image. There are two types of removal: page-level removal and zone-level removal. Page-level removal erases the background on the entire image page and overwrites the erased image with the new removed image for each recognition zone. Only image data for recognition zones is pasted into the new image. This reduces the physical size of the image on disk.

Zone-level removal removes lines and text that surround a recognition zone, and can also apply image filters for removing image noise and smoothing dot matrix print. This enables the recognition engine to process the image without extraneous information impacting the recognition results.

When you apply zone-level Image Enhancement functions to your image, the zone is registered and deskewed according to the page-level anchors and any zone-specific deskew settings you defined. Image Enhancement also delete all image data that is not contained within a zone from the processed image.

The following sections contain information on working with page-level and zone-level removal.

Setting and Testing Page-Level Zone Removal

Page-level zone removal uses the recognition zones as a template and, optionally, removes everything that you have not defined as a recognition zone on the image. To set and test page-level zone removal options:

1. Open a job in Designer and display the Form Properties, Image tab.

2. Select **Enhance** button to test both Enhancement and IPP settings on your setup image. The enhanced image displays in the image viewer.

3. To inspect all fields in your image, right-click on the image and hold the button to move the image around in the view area to see all your fields.

4. To accept the enhanced image, select **OK**. The new image does not overwrite the original image. Instead, the new image is saved to the \FormWare\<Image Path>.enh folder.

5. To exit the Image Enhancement test without saving the image, select **Cancel** at the bottom of the viewer.

Creating Removal Zones

You can specify removal parameters based on individual zones on the image. Zone-level removal removes horizontal and vertical lines from the zones on the form for improved recognition. By default, this type of removal does not alter the original image. Instead, the lines are removed in a processed version of the original image.

To set and test zone-level removal options for a field:

1. Select a form.

2. Double-click on the field you want to enhance to display the Field Properties dialog.

3. Select the IPP tab.

4. Select the **Enhancement** button. The Removal Features dialog displays. The X and Y coordinates of the zone display in the status bar.

5. Maximize the window.

6. Use the **Zoom In** and **Zoom Out** buttons on the toolbar to enlarge or reduce the field zone size.

7. If removal features are not set for this zone, click the **AutoFind** toolbar button. The tool automatically searches for horizontal and vertical lines and identifies them in blue in the tool image window. The **Next** button becomes active if more than one removal feature exists for the current zone. If you have previously set removal features for this zone, those settings display.

8. If a removal feature deletes valid data from to the zone, click the **Delete** button to delete the current removal object. Verify that the removal object you want to delete is the current object that appears in the **Type** field. The current object appears in blue on the image.

9. To delete all removal features for the current zone, click the **Delete All** button. A dialog displays asking you to confirm the deletion. Select **OK**.

10. Use the buttons on the toolbar to create new removals. To create a removal, select the button and then use the mouse to draw a zone around the line or text you want to remove. Buttons are:
    o **Select** - Access a removal feature, resize, or move the removal search area.
    o **Horizontal Line Tool** - Define a removal zone for a horizontal line. If the line is correctly detected, it will appear blue within the removal zone.
    o **Vertical Line Tool** - Define a removal zone for a vertical line. If the line is correctly detected, it appears blue within the removal zone.
    o **Vertical Tick Tool** - Draw a removal zone around two adjacent tick marks. Before selecting the tick tool, remove any horizontal lines that are near the tick marks and draw the initial zone to encompass exactly two tick marks. After you draw the initial zone, stretch the zone to encompass the width of the entire field, set the instances property to the number of tick marks, retest, and all tick marks are then removed.
    o **Text Tool** - Draws a removal zone around an area of text you want removed from the image.

11. Select the **Test** button to view the removals. If you need to add any other removals, first select the **Edit** button, then select the button that represents the type of removal you want from the toolbar.

12. Select **OK** to accept and save your removal settings.

13. Repeat these steps on the other fields on the form.

14. Select **Save All** to save your removal settings.

Setting Additional Zone Removal Properties

You can further define the removal zones by setting additional properties. To set properties:

1. On the Removal Features dialog, create a removal zone.

2. Select the **Properties** button. The Removal Features Properties dialog displays. This dialog contains a different set of properties for each removal feature type. The Field tab is the same for all removal feature types. The Feature tab contains different properties for each removal feature type. The Instances tab is the same for the Horizontal Line, Vertical Line, and Vertical Tick types, but does not exist for the Text type.

3. On the Field tab, set the values in the following fields to set global filters:

   a. In the **Point Size Filter** field, select the minimum point size of characters allowed in the field. A "point" is 1/72 inch. This filter is intended to remove small, isolated boilerplate text and other small marks within a zone. The default value for this field is 1. For 200 DPI images, or images with small print and lowercase characters, you may need set this to a value of 2 or 3. Set the size of the largest characters or marks to be removed so that extraneous data is removed, but valid text is not.

   b. In the **Width** and **Height** fields in the Noise Pixel Filters area, select the maximum size of *noise* fragments to automatically remove from the zone. The default is 3 by 3. Any speckles less than or equal to 3 pixels by 3 pixels are deleted from the zone. These settings remove dots throughout the zone, even in the middle of characters, if the dots are disconnected from the main character image.

4. Select the Feature tab, to indicate the variations of a feature that will be removed. Depending upon the type of feature you selected (i.e., horizontal line, vertical line, vertical tick, or text), set the appropriate field values.

Caution: Use caution when you have two adjacent fields and are attempting to remove marks in between them. Otherwise, you could inadvertently remove data characters.

Horizontal Line Feature

This tool enables you to remove lines above and below data. Using the property dialog, you can set parameters to remove multiple horizontal lines.

   o In the **Erase** area, select the boundary for removing black pixels: **Line Only**, which removes only the line detected; **Above Line**, which removes any black pixels including additional marks, characters, or noise from above the line, or **Below Line**, which removes any black pixels including additional marks, characters, or noise from below the line.

   o In the **Max Deviation** field, select the maximum number of pixels of noise to remove from the edge of the line. When a data character intersects the specified line, it must extend from the line by more than the defined number of pixels to preserve the intersecting portion of the character. Increase the value if you want to preserve more of the intersecting portion of the character.

   o In the **Line Thickness** field, select the expected line thickness of the removal feature to be detected (in pixels).

   o In the **Confidence** field, type the minimum accuracy level the removal feature must be confident of before it performs the removal process. This value is a percentage of 100.

Vertical Line Feature

The following removal features are available for a vertical line:

   o In the **Erase** area, select the boundary for removing black pixels: **Line Only**, which removes only the line detected; **Left of Line**, which removes all black pixels including additional marks and characters from the left of the line; or **Right of Line**,

---

which removes all black pixels including additional marks and characters from the right of the line.

- o In the **Max Deviation** field, select the maximum number of pixels of noise to remove from the edge of the line. When a data character intersects the specified line, it must extend from the line by more than the defined number of pixels to preserve the intersecting portion of the character. Increase the value if you want to preserve more of the intersecting portion of the character.

- o In the **Line Thickness** field, select the expected line thickness of the removal feature to be detected (in pixels).

- o In the **Confidence** field, type the minimum accuracy level the removal feature must be confident of before it performs the removal process. This value is a percentage of 100.

Vertical Tick Feature

Vertical ticks, or combs, are vertical lines connected by a right angle to a horizontal line. Vertical ticks divide data characters on the paper form. The following removal features are available for a vertical tick:

- o In the **Max Deviation** field, select the maximum number of pixels of noise to remove from the edge of the vertical tick. When a data character intersects the specified tick lines, it must extend from the lines by more than the defined number of pixels to preserve the intersecting portion of the character. Increase the value if you want to preserve more of the intersecting portion of the character.

- o In the **Line Thickness** field, select the expected line thickness (in pixels) of the vertical tick.

- o In the **Confidence** field, select the minimum accuracy level the removal feature must be confident of before it performs the removal process. This value is a percentage of 100.

Text Feature

The text removal feature refers to unwanted text characters, such as boilerplate, small print, labels, instructions, or extraneous information. The following removal features are available for text characters.

- o In the Erase area, select one of the following options for removing black pixels outside the boundaries of the removal feature: **Text Only**, which removes only the text characters detected; **Above Text**, which removes all black pixels and additional marks and characters from above the defined characters, or **Below Text**, which removes all black pixels and additional marks and characters from below the defined characters.

- o In the **Confidence** field, select the minimum accuracy level the removal feature must be confident of before it performs the removal process. This value is a percentage of 100.

- o In the **Text Height** field, select the average height in pixels of the text characters to remove.

5. Select the Instances tab, to determine how many occurrences of a feature are removed from the zoned area. This tab displays only for the horizontal line, vertical line, and vertical tick. It is not available for the Text feature. The **Instances Found** field, displays the number of marks found within the zoned removal search area

6. In the **Instances to Find** field, select the number of expected marks to remove within the zoned removal search area. For example, to remove five lines from the search area, define

one line and set this value to 5. Then select and drag the search zone to include the other lines. For vertical tick removal, this specifies the number of ticks to remove from the field.

Note: Tuning these parameters is a statistical process. There can always be a data form for which a certain set of parameter settings fails.

7. Select **OK** to return to the Removal Features dialog.

8. Select **OK** to return to the form.

9. Select **Save All** to save your settings.

Testing Zone Removal

After your removal feature test is successful at the zone level, you should test removal features against the entire form and job. You can test both registration and removal by selecting the **Enhancement** option on the Test tab on either the Job Properties or Form Properties dialog. Registration and removal are just two of a number of processes that you should test at the job, form, and field level.

Note: You should test and tune removal features on a test suite of images, not just on the Setup Image, to optimize removal performance across a broad range of input images.

Setting Form Removal Properties

Form Removal combines the image pre-processing (IPP) functions of cropping, deskew, registration, and line removal with a template-based "background removal" technology. On certain types of forms, removing all but the data you want to capture from a form prior to recognition can be a very effective, fast, and accurate way to process your forms. This type of technology will work effectively on:

- Simple non-dropout fixed forms (with originals scanned in from a scanner)
- Collections of non-dropout fixed forms (with originals scanned in from a scanner)

Form Removal will not work well on:

- Photocopied forms
- Faxed forms
- Forms that contain many different variations (pre-printed form text or graphics in different fonts or locations)

Background removal uses the process of comparing a filled form to a blank form (template) and determining the data that has been added (i.e., filled in). This is known as form extraction and is part of the Form Removal methodology.

Form Removal is a three-phase process:

1. Job-level IPP crops, deskews, despecks, and (optionally) registers all input images.

2. Form-level IPP (during the Enhancement process) performs horizontal/vertical line removal, character smoothing, deshading, inverse text, and image cleanup.

3. Form-level IPP (during the Enhancement process) then uses Form Removal to perform background removal. The background removal process uses the form Setup image as a "removal template," which is used internally to erase all portions of the image that display on the template. Effectively, this will leave only the data on the image that must be captured

during the recognition process. This third step is the process that can, in many cases, replace the normal Zone Removal process used in most FormWare applications.

Note: In cases where Form Removal is not completely successful, you can apply Zone Removal after Form Removal.

The removal template is the portion of the image that will be removed from images during production. Therefore, it is essential that the setup image used for Form Removal be a blank, unfilled image (i.e., no data can be in any field that you want to extract). Otherwise, you will be removing data from the image that you want to extract during recognition.

If you choose to use the Form Removal process, your setup image must be a blank, unfilled image that has been properly pre-processed and cleaned up prior to use.

The Form Removal process occurs as part of the FormWare Enhancement component. Therefore, certain settings you select within Designer indicate to the program that you plan to use Form Removal and tell the Enhancement component to process your forms in this manner.

Choosing Form Removal is as simple as selecting an additional check box option:

- If you check only the Remove background option on the Form Properties, IPP tab in Designer, FormWare will assume you plan to use only Zone Removal functions.
- If you check both the Remove background option and the Use setup image as form removal template option on the Form Properties, IPP tab in Designer, FormWare will assume you plan to use Form Removal functions. If you need to, you may still also use Zone Removal functions.

Setting Job-Level IPP Properties for Form Removal

To set job-level IPP properties for Form Removal:

1. Open Designer and create a new job with at least one form that will use Form Removal.

2. Display the Form Properties, Image tab to select a setup image for the form.

3. Select the **Browse** button adjacent to the **Setup Image** field to display the Browse Files dialog and the contents of the \FormWare\Images folder.

4. Highlight a setup image in the list and select **Open**. The file you selected is copied to the \FormWare\Forms folder and renamed to the same name as the current form with a .sup extension. This file name displays in the **Setup Image** field on the Form Properties, Image tab.

5. Check both the **Remove background** and **Use setup image as form removal template** check boxes on the Form Properties, IPP tab in Designer. This will automatically add default Job and Form IPP settings appropriate for Form Removal.

Note: The **Use setup image as form removal template** check box is disabled if a variant exists for the current form exists. The variant can be active or inactive.

6. You may skip all the rest of the following steps. If later testing (described in the next sections) shows the need, you can customize the default settings by following the steps below.

7. Display the Job Properties, IPP tab and the **Properties** button to set job-level IPP properties. The Image Pre-Processing Options dialog displays.

8. Check the following options on this dialog to enable job-level IPP functions for Form Removal:

   o **Image Deskew** - Applies the settings from the Image Deskew tab, which straightens the image on the page.

   o **Image Registration** - Applies the settings on the Horizontal Registration, Vertical Registration, and Registration Control tabs.

   o **Image Cropping** (optional) - Applies the settings on the Black Crop, White Crop, Border Crop, Miscellaneous Crop, and Manual Crop tabs.

Note: Do not enable or set horizontal or vertical line removal IPP properties at the job-level for Form Removal.

9. Select the **Properties** button to display the Image Pre-Processing Properties dialog.

10. On the **Deskew** tab, set properties that relate to image deskew. See the Setting Deskew Properties section for more information.

11. On the **Horizontal Registration** tab, set properties that relate to image registration. See the Setting Horizontal Registration Properties section for more information.

12. On the **Vertical Registration** tab, set properties that relate to image registration. See the Setting Vertical Registration Properties section for more information.

13. On the **Registration Control** tab, set properties that relate to image registration. See the Setting Registration Control Properties section for more information.

14. On the **Black Crop** tab, set properties that relate to image cropping. See the Setting Black Crop Properties section for more information.

15. On the **White Crop** tab, set properties that relate to image cropping. See the Setting White Crop Properties section for more information.

16. On the **Border Crop** tab, set properties that relate to image cropping. See the Setting Border Crop Properties section for more information.

17. On the **Misc. Cropping** tab, set properties that relate to image cropping. See the Setting Miscellaneous Cropping Properties section for more information.

18. On the **Manual Crop** tab, set properties that relate to image cropping. See the Setting Manual Crop Properties section for more information.

Note: Setting image cropping properties is optional for Form Removal.

19. Select **OK** on each IPP Properties tab before you exit.

20. Select **OK** on the IPP Options dialog to return to Designer.

21. Select **Save All** on the Designer window toolbar to save your changes.

Pre-Processing the Setup Image Into the Form Removal Template

Whether or not you are using Form Removals, the setup image must be pre-processed by job-level IPP to precisely deskew and match it more closely to the production-time images. To do this:

1. Display the Form Properties, Image tab for the job and form that will use Form Removal.

2. Select the **IPP** button. The IPP processes you set at the job-level run against the setup image selected for the job. Results of the test are displayed in an Image Viewer window. The processed image is on the left, the original image is on the right.

- Select **OK** to save the IPP changes to the setup image.
- Select **Cancel** to reject these changes. Return to the Job Properties, IPP tab and adjust the IPP properties you set on the Image Pre-Processing dialog for the job-level.

Note: If you have the **View Tests in Completion** button selected on the Designer window, only the processed image will display in a Completion window when the test is run. Therefore, for the purposes of this test, deselect this button before you run the test.

3. Continue to test these IPP settings until you get the desired result.

4. Select **Save All** on the Designer window toolbar to save your changes.

Defining Form Removal Properties

To set and test remaining properties for a job and form that uses Form Removal:

1. Display the Form Properties, IPP tab for the job and form that will use Form Removal. Both the **Remove background** and **Use setup image as form removal template** options are selected.
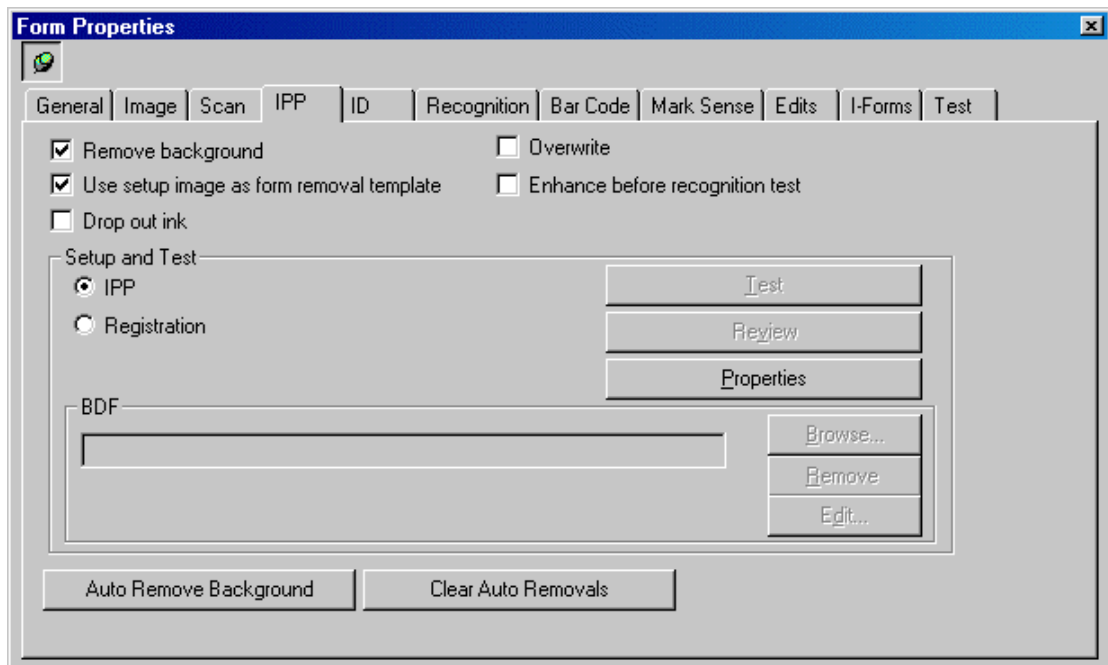
2. Return to the Designer window and define recognition zones for your data form. See the Defining Recognition Zones section for more information.

3. Display the Form Properties, Image tab to select a test image that contains data. This will allow you to test the recognition process against an image with data, since your current setup image contains no data.

4. Select the **Browse** button adjacent to the **Test Image** field to display the Browse Files dialog and the contents of the \FormWare\Images folder.

5. Highlight an image in the list and select **Open**. The file you selected is copied to the \FormWare\Forms folder and renamed to identify it as a test image for this form. This file name displays in the **Test Image** field.

Note: The setting in the Active Image area indicates the current image (Setup or Test) displayed in the image viewport in Designer. When you are using both a setup and a test image in your job, this allows you to switch between these two images for testing purposes.

---

6. Test recognition zones on your data form. See the Testing Recognition section for more information.

7. When the recognition tests are successful, select **Save All** on the Designer window toolbar to save your changes.

Testing a Single Image for Form Removal

To test a single image for Form Removal processing:

1. Open a job and form in Designer that is enabled for Form Removal (i.e., has a form removal template as its setup image).

2. Display the Form Properties, Image tab for this form.

3. Select **Browse** adjacent to the **Test Image** field and locate test image that contains data and has been processed through job-level IPP.

4. On the Form Properties, Image tab, select the **Enhance** button. This runs the Enhancement process over the test image. The results are displayed on the screen.

5. Examine the entire enhanced test image. Confirm that all lines, shading, inverse text, despecking, and form removal template processing was performed properly.

6. If the Enhancement process was not successful, it could be caused by one of the following:
    o   Faulty registration by the Enhancement process.
    o   Faulty IPP processing at the form level.
    o   Faulty Form Removal template processing. (This is most likely caused by the test image containing variations on the fonts or locations of the setup image's pre-printed text or graphics.)

7. To determine if the Form Removal process was faulty, display the Form Properties, IPP tab and deselect the **Use setup image as form removal template** option.

8. Display the Form Properties, Image tab and select the **Enhance** button. Examine the enhanced image. This will show the results of Enhancement registration and form-level IPP processing without Form Removal template processing.

9. If the test still shows errors, disable the form-level IPP processing by displaying the Form Properties, IPP tab and selecting the **Properties** button.

10. On the Image Pre-Processing Options dialog, deselect all options on this dialog and then select **OK**.

11. Display the Form Properties, Image tab and select the **Enhance** button. Examine the enhanced image. This will show the results of Enhancement registration processing without form-level IPP (line removals) or Form Removal template processing.

12. If errors occur at this point, the problem is with Enhancement registration. See the Registering an Image section to reset properties for image registration within your job.

---

13. Display the Form Properties, Image tab and select the **Enhance** button.

14. Continue to adjust and test image registration settings until no errors exist.

15. Display the Form Properties, IPP tab and select the **Properties** button.

16. On the Image Pre-Processing Options dialog, enable the form-level IPP processing options that you disabled in step 10.

17. Retest the Enhancement process on test image and, if you get errors, adjust the form-level IPP property settings until the Enhancement test is successful.

18. Display the Form Properties, IPP tab and select the **Use setup image as form removal template** option to enable Form Removal.

19. Retest the Enhancement process on the test image and, if you get error, you can use Zone Removals to clean up the remaining form items.

Testing a Batch of Images for Form Removal

To test a batch of images of a single form type for form removal:

1. Perform all steps in the Testing a Single Image for Form Removal section until you produce a clean test.

2. Open a job and form in Designer that is enabled for Form Removal (i.e., has a form removal template as its setup image).

3. Display the Form Properties, Test tab.

4. In the BDF area, select **Browse** to locate a BDF file that contains a batch of images to test.

5. On the Form Properties, Test tab, select the following options for the test (deselect all others):
   o **IPP**
   o ID (Form ID)
   o Enhancement

6. Select the **Test** button to run the test.

7. View your results to confirm that all images were processed correctly.

8. If some of the images were not processed correctly, perform the following steps:

   a. Deselect the **ID** and **Enhancement** check boxes, and then retest by selecting the **Test** button. This will indicate if the job-level IPP processing is in error. See the Setting Job-Level IPP Properties for Form Removal section for instructions on how to access these settings to fine-tune them.

   b. Reselect the **ID** checkbox and then retest by selecting the **Test** button. This will indicate if Form ID processing is in error. Fine-tune your Form ID settings and retest until your results are successful.

   c. Reselect the **Enhancement** checkbox and then retest by selecting the **Test** button. This will indicate if Enhancement processing is in error. Fine-tune your Enhancement settings and retest until your results are successful.

9. Select **Save All** on the Designer window toolbar to save your changes.

10. Repeat the steps in this task to test a batch that contains a mixture of different form types for Form Removal processing.

11. Select **Save All** on the Designer window toolbar to save your changes.

Testing Enhancement

You can test Image Enhancement settings at the job, form, and field levels using discrete or comprehensive testing. A discrete test tests just the Enhancement settings for the job (and includes the form and field settings). A comprehensive test tests the Enhancement settings along with other tasks and against either a batch of images or a single image.

When you conduct a comprehensive test, the BDF file must contain the JOBNAME=<jobname> parameter, where <jobname> is the name of the current job. Without this parameter, the comprehensive test will fail.

You can perform the following tests on your Enhancement settings:

- To perform a comprehensive test for a job, select the **Enhancement** check box on the Job Properties, Test tab.

- To perform a comprehensive test for a form, select the **Enhancement** check box on the Form Properties, Test tab.

- To perform a comprehensive test on a field, select the **Enhancement** check box on the Field Properties, Test tab.

- To perform a discrete test on your setup image at the form level, select the **Enhance** button on the Form Properties, Image tab.

- To perform a discrete test on the job at the field level, select the items you want to test on the Field Properties, IPP tab and select the **Test** button. Before selecting the **Test** button on the Field Properties, IPP tab, you must assign removal features.

- To perform a discrete test of removal features at the field level, access the Removal Features dialog from the Field Properties, Enhancement tab.

After you have defined registration anchors and removal features on your form within Designer, test these settings at the field level using the appropriate field-level tests. After your field-level tests are successful, you should test the registration and removal features against the entire form and job.

You should always test on a large batch of images to get a more reliable idea of how your settings will impact your job. Run at least a hundred images through testing. The extra time and care taken with the design process will pay off with fewer rejected forms.

Defining Recognition

Recognition is the process of identifying patterns on an image and interpreting those patterns into characters and other data. After you create the fields on the data form and properly register and deskew the image, you can create recognition zones. The type of data expected is specified on a field level, and recognition engines perform recognition on these fields.

When using recognition technology, you create recognition zones on the setup image that correspond to particular fields on the data form. A recognition engine attempts to recognize and insert data into each field on the data form based on the area you zoned on the setup image. During a Completion session, an operator views the contents of each field to verify recognition accuracy. Here the operator can fix unrecognizable or questionable characters within the fields, ensuring the most accurate results.

While no recognition process is 100% accurate, the ultimate objective in using recognition technology is applying the appropriate types of recognition to the image and achieving the most effective results. A common misconception about recognition is that a well-designed paper form guarantees recognition accuracy. This is not true, because not all paper forms are designed for recognition, and may paper forms cannot even be scanned.

Within a FormWare forms processing application, the recognition process is performed after Scanning, Image Pre-Processing, Image Enhancement, and Form Identification are complete. These processes occur before recognition for optimal recognition accuracy and acceptance. Use application-specific validation rules to optimize acceptance and accuracy in addition to using the built-in features.

Recognition engines read the machine print, handprint, check box marks, or bar codes on images and convert them into data. The FormWare recognition process accepts image files – both in batches and on an individual transaction basis – and converts the recognized data on the images into field-specific text. The resulting output file (*.OCR file) is a binary file that contains the text for the fields of the data form as well as alternative recognition results for those fields. This file is then verified, repaired, and validated by an operator using the Completion module.

Recognition results are typically a first or best result, and an alternate result. The best result is the character on which the recognition engine places the highest level of confidence. The alternate result is the next choice. Confidence refers to a measure used to determine whether to reject or accept the character. Some recognition engines offered with FormWare place a confidence level on both the best and second results characters. The difference between the confidence of the best result and the confidence of the alternate result is the spread. On a scale of 0 to 100, the higher the confidence, the more assurance can be placed on the best result. This helps to determine if a character is substituted or rejected.

Substitution occurs when an incorrect character is output for the best result. Rejection occurs when the recognized character is determined to be below its confidence threshold. In the Designer module, you can tailor character acceptance and accuracy to your needs by setting a threshold for confidence. Within the Completion module, the operator corrects rejected characters that appear in the out-of-context reject ticker tape.

Understanding Recognition Types and Properties

Recognition types guide the recognition engine that is determining the kind of data expected from a specific area on the image. Specifying a recognition type increases the accuracy of the recognition process. The FormWare recognition process supports the following recognition types:

- **None** - No recognition is applied, and the forms processing application assumes that data capture is performed by keying from image.

- **Machine Print** - Recognizes machine-printed alphanumeric characters that have consistent, predictable shapes including fixed pitch, variable pitched, and kerned fonts. This recognition type is also referred to as Optical Character Recognition (OCR).

- **Handprint** - Recognizes alphanumeric characters that vary in shape, such as hand-printed characters. This recognition type is also referred to as Intelligent Character Recognition (ICR).

- **Mark Sense** - Recognizes checkmarks, Xs, or other marks placed in check boxes. This recognition type is referred to as Optical Mark Recognition (OMR).

- **Bar Code** - Recognizes industry-standard bar codes. Bar codes are comprised of self-contained information encoded in the widths of printed bars and spaces.

- **Automatic** - Determines whether the characters are machine print or handprint, then applies that recognition type.

- **Cursive** - Recognizes handwritten characters.

The following table displays recognition property options available from the Designer module. Each column represents a recognition engine currently supported by FormWare. The columns indicate whether the recognition engine supports the property listed on the left.

The following are recognition engine abbreviations:

INS =	InScript	PAR =	Parascript
INS2 =	InScript2	RSP =	RecoStar Professional
AEG =	AEG	CGK =	CGK
NST =	Nestor	OMR =	Mark Sense
MIT =	Mitek	BAR =	Bar Code

	INS	AEG	NST	MIT	PAR	CGK	OMR	BAR
Field Properties - General								
Type: Table Zone	Yes	Yes	Yes	Yes	Yes	Yes	No	No
Field Properties - Image								
Zone X	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Zone Y	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Zone W	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Zone H	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Char W	Yes	No	No	No	No	No	No	No
Char H	Yes	No	No	No	No	No	No	No
Char Gap	No	No	No	No	No	No	No	No
Line Gap	No	Yes	Yes	Yes	No	No	No	No

	INS	AEG	NST	MIT	PAR	CGK	OMR	BAR
Field Properties - Recognition								
Confidence Level	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Characters Per Inch	Yes	Yes	Yes	Yes	No	Yes	No	No
Errors Allowed	No	No	No	No	No	No	No	No
Data Offset	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Data Length	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Point Size	Yes	Yes	Yes	Yes	Yes	Yes	No	No
OCR Font: REG	Yes	Yes	Yes	Yes	Yes	Yes	No	No
OCR Font: OCRA	No	Yes	Yes	Yes	No	Yes	No	No
OCR Font: OCRB	No	Yes	No	No	No	Yes	No	No
OCR Font: MICR	No	Yes	Yes	Yes	No	Yes	No	No
OCR Font: Farrington	No	Yes	Yes	No	No	Yes	No	No
Read Format: Word	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Read Format: Line	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Read Format: Paragraph	Yes	Yes	Yes	Yes	No	Yes	No	No
Recognition Type: Machine Print	Yes	Yes	Yes	Yes	No	Yes	No	No
Recognition Type: Handprint	Yes	Yes	Yes	Yes	Yes	Yes	No	No
Recognition Type: Bar Code	No	No	No	No	No	No	No	Yes
Recognition Type: Mark Sense	Yes	Yes	Yes	Yes	No	Yes	Yes	No
Recognition Type: Automatic	No	Yes	No	Yes	Yes	Yes	No	No
Recognition Type: Cursive	No	No	No	No	Yes	No	No	No
Segmentation	No	No	Yes	No	No	No	No	No
Preserve Punctuation	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Read length	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Dictionary (all properties)	Yes	No	Yes	No	Yes	Yes	Yes	No
Field Properties - Bar Code								
All Properties	No	No	No	No	No	No	No	Yes
Field Properties - Mark Sense								
Single Box	Yes	Yes	Yes	Yes	No	Yes	Yes	No
Multi Box / Multi Setup	Yes	Yes	Yes	Yes	No	Yes	Yes	No
Character If Sensed	Yes	Yes	Yes	Yes	No	Yes	Yes	No
Character If Not Sensed	Yes	Yes	Yes	Yes	No	Yes	Yes	No
Threshold levels (all properties)	Yes	No	No	No	No	No	Yes	No
Non-Dropout boxes	Yes	No	No	No	No	No	Yes	No
Field Properties - Check box Edits								
Zero Fill If Data	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Zero Fill No Data	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Right Justify	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

	INS	AEG	NST	MIT	PAR	CGK	OMR	BAR
Field Properties - Selective Edits								
Range (all properties)	Yes	Yes	Yes	Yes	No	Yes	Yes	No
Rule Name / Rules	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Form Properties - Recognition								
Alternate Characters	Yes	Yes	Yes	Yes	No	No	No	No
Job Properties - Recognition								
Voting (all properties)	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes

Setting Recognition Job Properties

When you create a forms processing application using the Designer module, you also select properties that affect the Recognition process. As with other processes, you need to set defaults at the job level. When you created the job, you set system defaults. Every new job inherits those default settings. Of course, you can modify the default settings for a job, form, or field.

To set Recognition properties for a specific job:

1. Open the job in Designer.

2. Select the **Properties** button to display the Job Properties, Recognition tab .

3. In the **Data Source for** field, select the file format of the recognition results file for the job. Options are:

    o **Recognition Format** - Indicates to Completion that a standard binary recognition results file exists for each image.

    o **Separate File** - Indicates to Completion that a single text recognition results file exists for each image.

    o **Image File** - Indicates to Completion that the recognition results have been inserted into the image file header.

    o **Batch File** - Indicates to Completion that a single text recognition results file exists for an entire batch.

4. In the **File extension** field, type a new value for the file extension of the recognition results files. The default extension for all recognition results files is .OCR. However, if the recognition results files for this job should have a different extension, type that extension in this field. If your recognition results files do not have extensions, type "…" in this field.

5. In the **Error character** field, type the character that represents a single reject in a data field during Completion. A "?" is the default character. If you select a different error character, ensure that you have chosen a character that will never be a valid data character. Otherwise, Completion will stop on those characters and treat them as errors.

6. In the **Correction character** field, type the character to use when an entire field has been rejected and needs to be re-keyed by the operator. The "~" is the default character.

7. In the **Data size** field, select the total length of all the data in a recognition results file. You can use the up and down scrolling arrows on the field to increment the record size, or you can place the cursor in the field so the number can be entered from the keyboard.

8. Select the **Batch reject repair** check box to enable batch-based reject repair. This displays all reject characters for an entire batch in the reject ticker tape of the Completion module. The default behavior is to display the reject characters per record. Use this with the Batch Build JobFlow task.

9. Select the **Position on first error** check box to position the cursor on the first recognition error in the field during Completion. Once an operator has keyed and released the field, the cursor advances to the next field that contains an error.

10. Select the **Position on next error** check box to position the cursor on the first recognition error in the field in Completion. Once the operator has corrected the error, the cursor jumps to the next error, even if it is in the same field. Use this option with the **Position on first error** check box.

11. Select the **Only process errors** check box to skip any fields during Reject Repair for which recognition properties are not set. FormWare assumes that all fields contain recognition results unless you select this check box. No FPL or VBA edits will be executed for these fields. This also causes the field edit to be executed only on fields that contain rejects.

12. Select the **Read page data** check box to read data from the BDF. Up to 4000 characters per image can be read from a BDF. Data not appearing on the data form (e.g., a document tracking number or a claim number), but associated with the image, can be extracted from the BDF. The Scan module or the program then places this data in the BDF. Completion inserts the data in the data record, beginning with the first field and continuing for the length of the data associated with the image.

13. In the **Voting** area, set the values to determine the precedence of multiple recognition engines. Voting is the ability to perform recognition with multiple recognition engines. Algorithms (user-defined or FormWare-supplied) are used to compare the results from each engine and determine the best data. When performing recognition on an image, the software checks for the existence of recognition results (.OCR) file. If one exists and the voting parameters are selected, the software executes the voting algorithms against the data from the existing file and the data from the second engine. Based on the outcome of the voting, the software overwrites the existing recognition results file.

Note: If a recognition results file exists and the voting parameters are not selected, the recognition results file from the first engine is overwritten with the results from the second.

14. Select from the following options:
    o   **User defined DLL** - Performs voting using a user-defined DLL. This option is mutually exclusive from the other three. For a sample of this DLL file, refer to the Creating a Voting DLL section.
    o   **Select if match** - Accepts the character with the highest confidence level when each engine returns the *same* character and the confidence level from both engines exceeds the confidence level specified on the form.
    o   **Select highest confidence** - Accepts the character with the highest confidence level when each engine returns *different* characters and the confidence level for all characters exceeds the confidence specified for the field.
    o   **Select highest spread** - Selects the character that has the highest spread in confidence value between the primary character and the alternate character when the primary character's confidence level exceeds the confidence level specified for the field. This option is used with an engine that provides an alternate character.

15. Voting algorithms follow a specific sequential process when you select one or more algorithm options:

    a.   If the **Select if match** criterion is met during the voting process, the algorithm chooses the best character.

    b.   If the **Select if match** option is not met, the algorithm uses the **Select highest confidence** criterion.

    c.   If **the Select highest confidence** option is not met, the algorithm uses the **Select highest spread** criterion.

    d.   If the characters are returned from all three checks, the character with the highest confidence is placed in the recognition results file.

Note: The voting algorithms only apply to hand print and machine print fields. These algorithms are not used on mark sense, bar code, or cursive fields.

16. Select the **Save All** button on the Designer window toolbar to save your settings.

Setting Recognition Form Properties

Use the Form Properties dialog, Recognition tab to set recognition properties for the individual forms within a job. At the form level, you specify the number of alternate characters returned by the recognition engine, test form-level recognition, create a form-level dictionary, and set full text recognition properties.

To set Recognition properties for a specific form:

1. Open the job in Designer.

2. Double-click the form name in the Job Browser to display the Form Properties, Recognition tab.

3. In the Alternate characters field, select the number of alternate characters you want the recognition engine to supply as additional choices for any recognized data. Alternate characters are also used for voting or by field validation rules to select the most accurate field result. The PerfectPost engine uses alternate characters to perform fuzzy address lookups. This field is not applicable to Full Text Recognition jobs.

4. Select the Create Form Dictionary button to create a form-level dictionary for use by the CGK recognition engine. A form level dictionary contains values from all of the dictionaries assigned to fields on the current data form. The values contained within this dictionary are compared to recognition results to provide better accuracy.

5. Select the Full Text Properties button to display the Form Properties dialog where you set parameters to convert TIFF images of documents into an editable format. Full Text Recognition uses an Optical Character Recognition (OCR) engine to convert scanned documents into word processing, spreadsheet, or database files. By using this process, you can eliminate time consuming, physical retrieval of boxed and stored files by archiving documents electronically.

Note: Setting properties at the form level assumes that you have previously set full text recognition properties at the job level (see previous section).

6. Select the **Full Text Recognition** check box to identify this form as a full text capture form. This launches the full text recognition engine over this form during the recognition process.

7. Select the **Default to job properties** check box to use the Unidentified Forms properties set at the job level. Use this option when you are creating a full text and data capture job.

Selecting this option disables the image processing options on this dialog and uses the options set at job level, as this form does not use Form ID.

8. In the **Form** area, select one or more of the following options for this full text form:

   o **Poor Image** - Slows the OCR processing to improve the processing of poor quality images for this full text form.

   o **Dot Matrix -** Enables a special recognition engine to process dot matrix print for this full text form.

   o **News Print** - Enables a special recognition engine to process newspaper text for this full text form.

   o **Fax** - Enables a special time stamp read engine for this full text form.

9. Select **Apply** and then **OK** to save your changes and return to the Form Properties, Recognition tab.

Selecting a Recognition Engine

FormWare provides a number of recognition engines that can identify the majority of the recognition types, along with some specialized recognition engines for specific recognition types, such as Bar Code or Mark Sense. However, no one recognition engine can necessarily obtain the highest recognition accuracy rate on every type of data. One type of engine may recognize handprint numeric characters better than another engine. Another engine may be better at recognizing unconstrained cursive alphanumeric characters. Choose a recognition engine that provides the best recognition accuracy rate on your image data.

You can specify one or more recognition engines on a per field basis. Following is a list of recognition engines available with FormWare and the types of recognition on which they are most effective:

- **AEG** - Effective on machine print recognition, including special fonts, dropout mark sense, out-of-context alphanumeric handprint, upper and lower case alpha, and numeric. The AEG engine also supports a number of foreign languages.

- **CGK** - Effective at recognizing contextual alphanumeric handprint, dropout mark sense, and machine print. You can use Trigram analysis and dictionaries with this engine. Unlike the other recognition engines that support field-level dictionaries, CGK creates a form-level dictionary. The CGK engine also supports a number of foreign languages.

- **InScript** - Effective at recognizing hand-printed numeric and uppercase alpha, hand and machine print dropout and non-dropout mark sense, and machine print. The InScript engine uses the Caere engine and neural network voting for machine print.

- **InScript2** - Effective at recognizing English and any other language based on the Hindu-Arabic character set, machine printed text, hand printed text, OCRA and OCRB machine printed text, MICR machine printed text, bar code machine printed characters (standard and 2 dimensional), and mark sense hand filled in data (unconstrained, or square, rectangular, circle, or oval constrained, with or without dropout boxes). InScript2 recognition integrates multiple recognition modules, which perform a variety of processing, one or more per print type.

- **Mitek** - Effective at recognizing dropout mark sense, out-of-context alphanumeric handprint, including upper and lowercase alpha. It also supports CAR and LAR recognition.

- **Nestor** - Effective at recognizing dropout mark sense and out-of-context alphanumeric handprint, including upper and lowercase alpha. It uses a version of the Caere engine for machine print. It can make effective use of dictionaries to improve recognition results.

- **Parascript** - Effective at recognizing upper and lowercase handprint and cursive print for fields that have a limited set of possible field values listed in a dictionary. Recognition performance is highly correlated with the quality, completeness, and sparseness of the dictionary. These include first names, last names, and addresses.

- **OMR** (Mark Sense) - Effective at recognizing single and multiple dropout and non-dropout mark sense only.

- **RecoStar Professional** - Incorporates the AEG and CGK recognition engines and includes its own internal voting combination process. RecoStar takes the best of both AEG and CGK recognition abilities, and through the use of an advanced engine configuration parameters dialog, enables you to select from voting accuracy bias settings to optimize recognition performance. The RecoStar client allows users to select a specific engine to use at the field level.

- **Bar Code** - Effective only on bar code recognition. This is the only FormWare-supported engine that recognizes bar codes. The Bar Code recognition engine is presented as a separate JobFlow task from the Recognition JobFlow task.

Installing Recognition Engines

When you purchase FormWare, you receive the InScript and InScript2 recognition engines as part of the Designer module. InScript and InScript2 are automatically installed when you install FormWare. This allows you to test recognition on the various parts of your forms processing application. Additional recognition engines must be purchased and installed separately.

The InScript and InScript2 engines require their own hardware key to run in a production environment. If you are using InScript recognition, the speed of the engine is controlled in the gray key. When your license file (DAT) contains a license for InScript, the recognition engine will run, but only at the slowest speed of 25 CPS. When InScript recognition is used with the gray key, recognition speed can be significantly increased to the optimum characters per day speed.

Many of the other engines also require their own hardware key to run both inside and outside of the Designer module. Remember to install the recognition engine hardware key on the machine that is running the recognition engine task. Do not attach the recognition key on the FormWare server machine.

Note: Using the Security License Manager, you can specify the recognition speed for that license during login. Previously, if two or more licenses were activated simultaneously for the same client but each license used a different recognition speed, the speed chosen was based on a "first come, first served" basis by the license manager. Now, you can request a speed by setting the FWSLM_OPTION=nnn INI property (nnn = speed, must be a number) for the client. The requested speed must exist in the DAT file and an available license at that speed must be present, otherwise the login will be denied.

Configuring a Recognition Engine

You can set advanced configuration properties for the AEG, CGK, and RecoStar recognition engines.

Note: You can also set advanced configuration properties for the InScript2 recognition engine. See the Configuring the InScript2 Recognition Engine section for more information.

1. Double-click a field on the form to display the Field Properties dialog.

2. Select the Recognition tab.

3. In the Recognition Engines field, select any or all of the following options: AEG, CGK, RecoStar.

4. Select the Advanced Properties button.

5. If you select more than one engine type that can be configured on the Advanced Properties dialog, the Configurable Engines dialog displays. Highlight an engine for which you want to define advanced properties and then select OK.

6. The Advanced Properties dialog for the selected engine displays.

Note: The Advanced Properties dialogs for the AEG, CGK, and RecoStar engines are similar but not identical. The steps that follow represent the options that could potentially display on any of these dialogs. Note that certain *engine types* or *recognition types* may not support certain options.

7. The recognition type you selected for the field displays in the **Recognition Type** field. The following recognition types can appear in this field: **Machine Print**, **Hand Print**, **Automatic**, and **Cursive**.

8. Select the **Multi-font Segmentation** check box to enable the recognition engine to better separate the characters in the zones, which contain the fonts of various shapes and sizes. This feature is particularly useful for Free Form recognition, where a single zone surrounds the whole page.

9. In the **Engine** area, select a specific sub-engine to use at the field level. Options are: **RecoStar** (default), **CGK**, or **AEG**.

Note: The **Engine** area, only displays on the RecoStar Advanced Properties dialog.

10. In the **Voting Accuracy Bias** field, select the level of voting accuracy you want to apply to this field. There are three options:
    o   **High** – Bias towards accuracy, resulting in potentially more rejected characters.
    o   **Medium** – Accuracy and reject rate are balanced. This is the default setting.
    o   **Low** – Bias towards lower reject rate, resulting in lower accuracy of non-rejected characters.

11. Voting increases recognition accuracy and rejects characters on which the recognition engines disagree. When a recognition engine is executed to read an image, the recognition software checks for the existence of a recognition results (.OCR) file that was output from another recognition engine. If one exists and the voting parameters were specified for the image, the software applies voting algorithms against the data from the OCR file and the data from the second recognition engine. Based on the outcome of the voting, the software overwrites the existing recognition results file. This process continues until all selected recognition engines have completed processing.

Note: Because the RecoStar recognition engine actually performs a voting combination of the CGK and AEG engines during the recognition process, this option applies only to the RecoStar Pro recognition engine. This voting process is internal to the RecoStar engine and does *not* use the FormWare voting function.

12. In the **Measurement units** area, select the option that indicates the character spacing in the field. Options are: 1/**100 inch** or **1/100 mm**.

13. In the **Context Analysis** area, select the value that indicates the field content-specific information to the recognition engine. Values are:

   o **Word Type** – Field contains words of alpha and/or numeric characters.

   o **Trigram** – Recognition engine should apply a language-specific trigram table when determining reject characters. A trigram table provides all possible three-character combinations for words within the language being read. For example, when recognizing the word, NAME, the recognition engine will compare the first three characters NAM to the trigram table to determine whether the NAM combination of characters is logical. It will then compare the AME combination to determine its logic or likelihood. However, if the recognition engine recognized the letter A as an H instead, and compared the combination NHM, it would reject the characters based on the illogical combination and the absence of that combination in the trigram table.

14. In the **Machine Print** area, select the horizontal spacing of the machine print characters in the **Spacing** the field. This area is active only if you have selected either Machine Print or Automatic in the **Recognition type** field on the Field Properties, Recognition page. Spacing options are:

   o **Automatic** – Indicates that the engine determines whether the print is fixed or variable for the field. This is the default setting.

   o **Fixed unknown** – Indicates the print has fixed spacing for the field, but the specific character-to-character spacing is not known in advance.

   o **Fixed known** – Indicates uniform character spacing for the field. Selecting this option activates the **Character Spacing** field.

   o **Variable** – Indicates the print has proportional spacing for the field.

15. In the **Character Spacing** field, select the width of the spaces between characters, inches or millimeters (1/100th), depending on the selection in the **Measurement units** field.

16. In the **Hand Print** area, select the horizontal spacing of the machine print characters in the **Spacing** the field. This area is active only if you have selected either Hand Print, Automatic, or Cursive in the **Recognition type** field on the Field Properties, Recognition page. Spacing options are:

   o **Automatic** – Indicates that the engine determines whether the print is fixed or variable for the field. This is the default setting.

   o **Fixed unknown** – Indicates the print has fixed spacing for the field, but the specific character-to-character spacing is not known in advance.

   o **Fixed known** – Indicates uniform character spacing for the field. Selecting this option activates the **Character Spacing** field.

   o **Variable** – Indicates the print has proportional spacing for the field.

17. In the **Character Spacing** field, select the width of the spaces between characters, inches or millimeters (1/100th), depending on the selection in the **Measurement units** field.

18. In the **Field Orientation** area, select the option to determine this field's orientation on the data form when the recognition zone is rotated in some fashion. For example, when recognition is processing a zone printed along the vertical border of the page (at 90 degrees rotation). Options are:

   o **0 Degree** (default)

   o **90 Degree**

   o **180 Degree**

   o **270 Degree**

19. Select one of the following buttons to continue:
    o **OK** to accept your changes on this dialog and exit
    o **Cancel** to reject your changes on this dialog and exit
    o **Reset** to return all settings on this dialog to their defaults and exit

Changing Recognition Engines

After you create fields on the data form, you can change the recognition engine(s) assigned to those fields. You can do this on a per field basis by selecting an option in the **Recognition Engines** field on the Field Properties, Recognition page for each field. You can also quickly change the recognition engine(s) specified for a number of fields at the same time.

You can also use the Multi-Field Properties pages to change the recognition engine selected for multiple fields. See the Editing Properties for Multiple Fields section for more information.

Follow these steps to change the recognition engine assignment for one or more fields:

1. Within the Designer module, use the mouse to select one or more fields on the data form.

2. Right-click and select the Select Engine(s) option from the context menu. A list of all available recognition engines displays.

3. Highlight the desired recognition engine or hold down the [Shift] or [Ctrl] keys to select multiple engines from the list.

4. Select OK. All selected fields will use the specified engine(s) for recognition.

Configuring the InScript2 Recognition Engine

InScript2 performs recognition on machine print, handprint, OCRA, OCRB, MICR, bar codes, and mark sense data. "Auto" mode is not supported for auto-detect and recognition of machine print/hand print data on the same field.

You can enhance the performance of InScript2 recognition using dictionaries of words found within the documents, as well as any provision of lexical patterns (regular expressions) that are common within the documents. A Standard English lexicon is supplied, and custom user lexicons can also be supplied when required.

As with any recognition engine, InScript2 will generate errors during text recognition. You can reduce the percentage of text recognition errors using settings that request a higher accuracy at the expense of processing time. Certain types of data have alternate InScript2 modules that perform at a higher rate of accuracy, but take longer to process. Post-recognition editing is available using the standard FormWare recognition client processing methods.

The purpose of the InScript2 recognition engine is to accurately capture full OCR file-compatible information for the following types of form data:

- English and any other language based on the Hindu-Arabic character set
- Machine printed text
- Hand printed text
- OCRA, OCRB machine printed text
- MICR machine printed text
- Bar code machine printed characters (standard and two-dimensional)

- Mark sense hand filled in data (unconstrained, or square, rectangular, circle, or oval constrained, with or without dropout boxes).

Note: InScript2 does not support Cursive recognition.

For most types of form data, InScript2 can support more than one recognition module. However, mark sense recognition has only one module. You can purchase InScript2 with licensing for all recognition features, or for two subsets of recognition features – character recognition (all types) and mark sense (all types), or bar code recognition only.

Using InScript2, you can select a specific recognition feature on a field-by-field basis, when more than one type of ScanSoft licensed feature is supported for the field's data type, and is licensed from ScanSoft. The different recognition features provide tradeoffs in speed, accuracy, and some functionality within the same general data type.

InScript2 includes:

- **InScript2 Recognition Engine** – InScript2 recognition integrates multiple recognition modules, which perform a variety of processing, one or more per print type. This integrated set of modules is called the InScript2 recognition engine.

- **InScript2 Recognition Client** – The InScript2 recognition client (In2Clnt) is the recognition client built on the base (generic) recognition client common to InScript, RecoStar Pro, AEG, CGK, etc. In2Clnt uses a special recognition engine development kit, which combines a number of specialized recognition engines, many of which have advanced tunable features available to the end user. The In2Clnt.ocx program is located in the \FormWare\Programs folder.

Within the InScript2 client, support is provided on a **field-by-field basis** to:

- Add a user dictionary
- Match literal items
- Match regular expression items
- Make speed/accuracy tradeoffs
- Select a language for the field's recognition

Support is provided on a **form-by-form basis** to:

- Make speed/accuracy tradeoffs for the form
- Select a language for the form's recognition (i.e., character sets)

Understanding InScript2 Recognition Modules

By default, all FormWare recognition engines support machine print, handprint, and mark sense. Additionally, bar code recognition is also supported within InScript2.

The following recognition modules are available with InScript2:

- **DOT** - Draft Dot-Matrix module
- **HNR** - Handprint numeric module
- **MAT** - Matrix Matching module
- **MOR** - Multi-Lingual OmniFont module
- **MTX** - OmniFont module
- **OMR** - Optical mark module
- **PLUS** - OmniFont module

- **RER** - Handprint alphanumeric module
- **SEA** - Seaport Bar Code module (1- and 2-dimensional)

When multiple modules are available for a particular field and recognition type, InScript2 makes a default selection during run-time or uses the module you select in Designer. When the default is not installed or licensed, a "cascading fallback" sequence of modules occurs during the recognition process. When no appropriate module is installed or licensed, a recognition test from Designer displays an error message or, during runtime, a fatal FDF setup error is reported during recognition processing and processing is aborted.

It is expected that all versions of InScript2 will be installed with all available recognition modules. The SEA module requires an additional license, but is installed by default to simplify the software installation process. Typically when multiple recognition modules are available and licensed, the default is to use the more "powerful" engine. Users can override this default and select the specific engine to use for each field.

Understanding InScript2 Recognition Types

InScript2 supports the following recognition types:

- Machine Print Recognition
- Handprint Recognition
- Mark Sense Recognition
- Bar Code Recognition
- OCRA and OCRB Font Recognition
- MICR Font Recognition

Understanding Machine Print Recognition

Machine print recognition recognizes machine-printed alphanumeric characters that have consistent, predictable shapes including fixed pitch, variable pitched, and kerned fonts. This is also referred to as OCR. InScript2 defaults to MTX for machine print characters. If MTX is not installed or licensed, it uses MOR. DOT is never used as the default for machine print. Within InScript2, the following recognition modules are available for machine print characters:

Dot Matrix – Used specifically for applications or fields that will always have low quality (i.e., 9 pin) dot matrix print. Use one of the other modules for near letter quality and letter quality dot matrix print.

OmniFont (MTX) – Supports full character sets for the following common languages: English, French, Spanish, Italian, German, Norwegian, Portuguese, Danish, Dutch, Finnish, Swedish, and Brazilian. This is the default machine print engine. This engine performs well for general recognition particularly when a good percentage of the forms contain lightly printed, broken, fragmented, or dot-matrix characters. The Fast setting uses alphanumeric recognition with coercion for numeric fields, is much faster than the Accurate setting, and is very close in accuracy to the accurate setting (which uses numeric recognition for numeric fields). A limitation of the MTX (M/Text) engine is that the confidence range for characters is very coarse - 0,75, or 100.

OmniFont (MOR) – Supports full character sets for all 114 languages available with InScript2. This recognition module is also trainable. The MOR engine is faster than the MTX engine. It also recognizes most languages, provides a better set of confidences, and is reputed to be more accurate on cleanly printed text. If you select the MOR engine, InScript2 will run the MTX (M/Text) engine if it detects that the page has broken Dot Matrix text. If a form has no validation

edits for certain fields and you are only relying on the confidence level to reject characters or fields, the MOR engine is a better choice.

PLUS (Voting) – Uses voting technology to provide the best result. This module combines the power of two other recognition modules – MOR and MTX. The accuracy of PLUS is considerably better than MOR or MTX separately (about 40% fewer errors), but the recognition needs more time (50% more than MTX). If you select the Plus engine, InScript2 will run the MTX (M/Text) engine if it detects that the page has broken Dot Matrix text.

Understanding Handprint Recognition

Handprint recognition recognizes alphanumeric characters that vary in shape, such as hand-printed characters. This is also referred to as Intelligent Character Recognition (ICR). InScript2 defaults to HNR for Unsigned Numeric and Signed Numeric fields. All other field types default to RER. Within InScript2, the following recognition modules are available:

Handprint (numeric only) – Used for numeric fields only (i.e., digits, and the following symbols '+', '-', '.', and ',').

See the Setting HNR (Handprint) Sub-Engine Properties section for instructions.

Handprint (RER reRecognition) – Used for full, multi-lingual alphanumeric handprint characters. This is a third-party recognition module and requires advanced settings on a field-by-field basis.

See the Setting RER (Handprint) Sub-Engine Properties section for instructions.

Understanding Mark Sense Recognition

Recognizes checkmarks, Xs, or other marks placed in check boxes. This recognition type relates to the industry term Optical Mark Recognition (OMR). No multiple sub-engine selection is available for the mark sense type.

Understanding Bar Code Recognition

Recognizes industry-standard bar codes. Bar codes are comprised of self-contained information encoded in the widths of printed bars and intervening spaces and can be divided into two types:

One-dimensional (1D) bar codes – The traditional bar codes that you see on product packaging and are used to encode short messages like part, stock, serial, or customer numbers.

Two-Dimensional (2-D) bar codes – Encode larger masses of data like the contents of an entire shipment or a portion of a patient's medical record. PDF-417 is the most popular 2-D bar code. The size of the PDF-417 is similar to a standard bar code, but by including another dimension, more information is encoded in the same space. In addition to the message, this bar code contains a significant amount of error correction code that will allow the message to be read even if part of the symbol is damaged or destroyed during handling. This type also supports up to ten bar codes per field.

Within InScript2, the Bar Code (SEA 1-,2- dimensional) recognition module is available for bar code characters. It is used for one- and two-dimensional types of bar codes – CODE 39 (Code 3 of 9), CODABAR, ITF (2 of 5 interleaved), CODE 128, Code 2 of 5, Code 93, UPC-A, UPC-E, EAN-13, Postnet, and the two-dimensional PDF-417 2D code.

The SEA engine can read bar codes at any angle and any size in addition to reading both one-dimensional and two-dimensional types of bar codes. Additionally, the SEA engine allows you to select the types of bar codes to recognize.

By default, the SEA engine can include up to 10 bar codes per zone. The zone can contain other elements besides the bar code. The program can detect the bar code(s) inside a zone. If there are other objects (text, figures, etc.) in the zone, recognition will be slower and these other elements in the zone will not be recognized. One-dimensional bar codes cannot be enabled with two-dimensional ones in a single zone.

See the Setting SEA (Bar Code) Sub-Engine Properties section for instructions.

Understanding OCRA and OCRB Fonts

OCRA and OCRB are two commonly used character-recognition "friendly" machine print fonts. InScript defaults to MAT for OCRA/OCRB. If MAT is not installed or licensed, then MOR, and then MTX are used (respectively). Within InScript2, the following recognition modules are available for characters in the OCRA and OCRB fonts:

OCR A/B and MICR – Specifically designed for older character-recognition "friendly" machine print fonts.

OmniFont M/Text – Supports OCRA and OCRB.

OmniFont MOR – Supports OCRA and OCRB.

Understanding MICR Font

MICR is a commonly used character-recognition "friendly" machine print font. No multiple sub-engine selection is available for MICR.

Selecting a Language for InScript2

InScript2 recognition supports 114 languages. If no specific language is specified for a field, it defaults to English. You can also specify a language at the form level on the Form Properties, General tab.

When you set this property at the form level, all unspecified fields on this form default to this language type. You can also override the language for a field by selecting a different language just for the specific field. on a field-by-field for the form by specifying a different language for a specific field.

Following is a list of the supported languages for InScript2:

Afrikaans	Albanian	Aymara
Basque	Bemba	Blackfoot
Breton	Bugotu	Bulgarian (Cyrillic) – also includes English language characters
Byelorussian (Cyrillic) – also includes English language characters	Catalan – spelling dictionary supported	Chamorro
Chechen	Chuana or Tswana	Corsican
Croatian	Crow	Czech – spelling dictionary supported
Danish – spelling dictionary supported	Dutch – spelling dictionary supported	English – spelling dictionary supported
Eskimo	Esperanto	Estonian
Faroese	Fijian	Finnish – spelling dictionary supported

French – spelling dictionary supported	Frisian	Friulian
Gaelic Irish	Gaelic Scottish	Ganda or Luganda
German – spelling dictionary supported	Greek – spelling dictionary supported – also includes English language characters	Guarani
Hani	Hawaiian	Hungarian – spelling dictionary supported
Icelandic	Ido	Indonesian
Interlingua	Italian – spelling dictionary supported	Kabardian
Kasub	Kawa	Kikuyu
Kongo	Kpelle	Kurdish language selection if written in Latin alphabet.
Lappish	Latin	Latvian
Lithuanian	Luba	Luxembourgian
Macedonian (Cyrillic) – also includes English language characters	Malagasy	Malay
Malinke	Maltese	Maori
Mayan	Miao	Minankabaw
Mohawk	Moldavian (Cyrillic) – also includes English language characters	Nahuatl
Norwegian – spelling dictionary supported	Nyanja	Occidental
Ojibway	Papiamento	Pidgin English
Polish – spelling dictionary supported	Portuguese – spelling dictionary supported	Portuguese (Brazilian)
Provencal	Quechua	Rhaetic
Romanian	Romany	Ruanda
Rundi	Russian (Cyrillic) – spelling dictionary supported – also includes English language characters	Samoan
Sardinian	Serbian (Cyrillic) – also includes English language characters.	Shona
Sioux	Slovak	Slovenian
Somali	Sotho, Suto or Sesuto	Spanish – spelling dictionary supported
Sundanese	Swahili	Swazi
Swedish – spelling dictionary supported	Tagalog	Tahitian
Tinpo	Tongan	Tun
Turkish	Ukrainian (Cyrillic) – also includes English language characters	Visayan
Welsh	Wend or Sorbian	Wolof
Xhosa	Zapotec	Zulu

Understanding Punctuation Characters Recognized by InScript2

The following table lists the punctuation characters recognized by the InScript2 recognition modules. The MOR, PLUS, and MTX recognition modules support most of these characters so they do not appear in separate columns (e.g., MTX does not support the Euro Sign). Additionally, most of the punctuation is language independent. The only exceptions are the Inverted Exclamation Mark and the Inverted Question Mark characters, which are validated only when the Spanish language is enabled.

Name of the Character	UNICODE	DOT	RER	BRA	HNR	MAT
Apostrophe-Quote	0x0027	DOT	RER			
Closing Curly Bracket	0x007d	DOT	RER			
Closing Parenthesis	0x0029	DOT	RER			
Closing Square Bracket	0x005d	DOT	RER			
Colon	0x003a	DOT	RER			
Comma	0x002c	DOT	RER		HNR	OCRA, OCRB
Double Quotation Mark Left	0x201c					
Double Quotation Mark Right	0x201d					
Em dash	0x2014					
Exclamation Mark	0x0021	DOT	RER	BRA		
Hyphen-Minus	0x002d	DOT			HNR	
Inverted Exclamation Mark	0x00a1	DOT				
Inverted Question Mark	0x00bf	DOT				
Low Dbl. Comma Quot. Mark	0x201e					
Opening Curly Bracket	0x007b	DOT	RER			
Opening Parenthesis	0x0028	DOT	RER			
Opening Square Bracket	0x005b	DOT	RER			
Period (full-stop)	0x002e	DOT	RER		HNR	All MAT
Question Mark	0x003f	DOT	RER			
Quotation Mark	0x0022	DOT	RER			
Semicolon	0x003b	DOT	RER			
Left Pointing Single Guillemet	0x2039					

Name of the Character	UNICODE	DOT	RER	BRA	HNR	MAT
Right Pointing Single Guillemet	0x203a					
Soft Hyphen	0x20ad					
Spacing Grave	0x0060	DOT				
MISCELLANEOUS CHARACTERS						
Ampersand	0x0026	DOT	RER			
Asterisk	0x002a	DOT	RER	BRA		OCRA
Backslash	0x005c	DOT	RER			OCRA
Bullet	0x2022					
Cent Sign	0x00a2	DOT				
Commercial At	0x0040	DOT				MICR
Copyright Sign	0x00a9					
Degree Sign	0x00b0					
Dollar Sign	0x0024	DOT	RER	BRA		OCRA
Equals Sign	0x003d	DOT	RER			
Euro Sign	0x20ac					
Greater-Than Sign	0x003e	DOT	RER			OCRA, OCRB
Left Pointing Guillemet	0x00ab	DOT				
Less-Than Sign	0x003c	DOT	RER			OCRA, OCRB
Number Sign	0x0023	DOT	RER			MICR
Percent Sign	0x0025	DOT	RER			
Plus Sign	0x002b	DOT	RER		HNR	OCRA, OCRB
Plus-OR-Minus Sign	0x00b1	DOT				
Pound Sign	0x00a3	DOT				
Registered Trade Mark Sign	0x00ae					
Right Pointing Guillemet	0x00bb	DOT				
Section Sign	0x00a7	DOT				
Slash	0x002f	DOT	RER			OCRA, OCRB
Sml. Script F	0x0192					
Space	0x0020	DOT	RER	BRA	HNR	all MAT
Spacing Circumflex	0x005e	DOT				
Spacing Underscore	0x005f	DOT	RER			
Tilde	0x007e	DOT				
Vertical Bar	0x007c	DOT	RER			
Yen Sign	0x00a5	DOT				

Setting InScript2 as the Default Engine

You can set InScript2 as the default recognition engine at the system, job, and field levels.

At the System Level

Follow these steps to use InScript2 recognition as the default for your FormWare system.

1. Open Designer.

2. From the Edit menu, select System Defaults. The System Defaults dialog displays.

3. Select the Form and Field tab.

4. In the Recognition engines field, select the InScript2 option.

5. Select Apply to apply your new system default settings.

6. Select OK to accept your changes and return to Designer.

7. Select Save on the Designer window to save your changes.

At the Job Level

Follow these steps to use InScript2 recognition as the default for your FormWare job. This job default setting overrides the system default settings for the current job.

1. Create a new job in Designer and select **New**. The New Job Name dialog displays.

2. Type the name of the new job in the **Job Name** field. The Job Defaults dialog displays.

3. In the **Recognition engines** field, select the InScript2 option.

4. Set the other options on this dialog as necessary for your job defaults.

5. Select **OK** to accept your changes and return to Designer.

6. Select **Save** on the Designer window to save your changes.

At the Field Level

Follow these steps to apply InScript2 recognition to a specific field within your FormWare job. This field setting overrides the system and job default settings for the field you specify.

1. Open the job and form in Designer.

2. Double-click on the field to display the Field Properties dialog.

3. Select the Recognition tab.

4. In the **Recognition engines** field, select the **InScript2** option.

   You can select multiple recognition engines in this field. If you select one or more engines that support advanced recognition properties, the Advanced Properties button activates and allows you to set additional recognition properties at the field level.

5. Select **Test** to test the InScript2 recognition process on this field.

6. Select **Save** on the Designer window to save your changes.

For Virtual Classification

You select and test a recognition engine for Virtual Form ID at the job level within Designer on the Job Properties, Test tab. You can use the following recognition engines for Virtual Form ID: InScript2, InScript, and RecoStar Professional.

> **Note**: You can use all other available and licensed recognition engines to test and perform recognition processing at the job level for an application that uses Virtual Classification, however, you can only use InScript, InScript2, or RecoStar for the Virtual Form ID process.

Follow these steps to setup and test Virtual Form ID on your FormWare job.

1. Open an existing job or create a new job in Designer.

2. Create at least two forms within the job and at least one field on each form.

3. Set up the job to use Virtual Classification.

4. Display the Job Properties, Test tab.

5. Select the **Recognition** check box.

6. In the BDF area, use the **Browse** button to locate a BDF to use in the test. Select the **Edit** button in this section to modify the contents of the BDF file, if necessary.

7. In the **Virtual Form ID Engine** field, select the **InScript2** option.

8. Select the **Test** button to test the InScript2 recognition process.

Setting InScript2 Form-Level Properties

Use the InScript2 Advanced Form Properties dialog to change the default settings for the InScript2 recognition engine at the form level. The settings on this dialog affect the properties of all InScript2 fields on your form.

InScript2 obtains the fastest recognition times when you use form-level settings instead of field-level settings. Field-level settings significantly slow down the system. For the fastest recognition times, ensure that your InScript2 engine is configured as follows:

*   Obtain a gray hardware key (dongle) for the machine that will be running InScript2.

*   On the InScript2 Advanced Field Properties dialog, ensure that the Accuracy Bias property is set to Fast and the Machine Print and OCRA, OCRB Preferred Sub-engines recognition modules are set to OmniFont (M/Text). By default, the Accuracy Bias Property is set to Fast, and the Preferred Sub-engines are set to OmniFont (M/Text).

Follow these steps to set advanced properties for InScript2 at the form level.

1. Open the job and form in Designer.

2. Display the Form Properties, General tab for the form.

3. In the Component field, select the InScript2 Recognition Engine option.

4. Select the **Configure** button next to the **Component** field. The InScript2 Advanced Form Properties dialog displays.

5. In the **Accuracy Bias** field, select from the following options. InScript2 applies the selected value to the language in the **Language** field.

    o   **Accurate** – Uses the most accurate processing method, resulting in potentially more rejected characters.

        o   **Fast** – Uses the fastest processing method. The processing bias is towards a lower reject rate, resulting in lower accuracy of non-rejected characters. This is the default.

6. In the **Language** field, select the language to use for this field. See the list of available languages in the Selecting a Language for InScript2 section. By default, this field is set to **English**.

7. Select the **Use Language Dictionary to Check Spellings** check box to use the selected language's dictionary to spell check words on the form. This option is active only if you have selected a language in the **Language** field and a separate dictionary is available for that language.

8. Select the **Use Language Dictionary to Correct Spellings** check box if you want to use the selected language's dictionary to correct misspelled words on the form. This option is active only if you have selected the **Use Language Dictionary to Check Spellings** check box.

9. In the **Preferred Sub-engines** area, select the values for the sub-engines you are using for a given type of data. For each given type of data a default is used when first defining field-level advanced settings or at run-time. If the preferred sub-engine is not available or acceptable, the selection you make determines the new default engine to use at the form level. For example, you may prefer handprint character type field to use the numeric-only engine, but force specific alphanumeric handprint fields to use the alphanumeric handprint engine. Set the following values:

        o   In the **Machine Print** field, select the engine to use as the default for this recognition character type at the form level. By default, this field is set to **OmniFont (M/Text)**.

        o   In the **Hand Print** field, select the engine to use as the default for this recognition character type at the form level. By default, this field is blank.

        o   In the **OCRA, OCRB** field, select the engine to use as the default for this recognition character type at the form level. By default, this field is blank.

10. In the **Sub-engine Configurations** field, select the appropriate sub-engine from the drop-down list to define additional properties, if available. The following options are available:

        o   **BarCode (Seaport 1,2 dimensional)**

        o   **Handprint (numeric only)**

        o   **Handprint (RER reRecognition)**

11. Select the **Configure** button to modify the settings for a sub-engine or select **Reset** to remove the sub-engine. See the Configuring InScript2 Sub-Engines section for more information.

Note: The changes you make to the sub-engine configuration will not apply unless you also select the same sub-engine in the Preferred Sub-engines area.

---

12. Select **OK** to accept your changes and return to Designer.

13. Select **Save All** on the Designer window toolbar to save your settings.

Setting InScript2 Field-Level Properties

Use the InScript2 Advanced Field Properties dialog to change the default settings for the InScript2 recognition engine at the field level. Field-level settings override the form-level settings. To set InScript2 at the field level:

1. Open the job and form in Designer.

2. Select a field on the data form or create a new one.

---

3. Set a recognition zone for the field on the setup image.

4. Display the Field Properties, Recognition tab for the field.

5. In the **Recognition engines** field, select the **InScript2** option. You can select more than one recognition engine for a field.

Note: If you select **InScript2** in the **Recognition engines** field and **Bar code** in the **Recognition type** field, only the **Test** button will be available on the Field Properties, Bar code tab. This occurs because you must set the properties for InScript2 Bar code recognition from the InScript2 Advanced Field Properties dialog.

6. Select the **Advanced Properties** button to display the InScript2 Advanced Field Properties dialog. The properties on this dialog override the advanced form-level properties and any field-level default settings related to the property you are changing and that exist for this field.

Note: The **Recognition Type** field displays the value from the **Recognition type** field on the Field Properties, Recognition tab. This value dictates the options that are available in the **Sub-Engine** field. Certain recognition types do not need or support the advanced property settings.

7. In the **Sub-Engine** field, select the sub-engine for this recognition type.

8. Select the **Disable Language Dictionary to Check Spellings** check box to disable the language dictionary spell checker option for this field. If you selected the **Use Language Dictionary to Check Spellings** check box on the InScript2 Advanced Form Properties dialog, checking this option *disables* the language dictionary spell checker for this field only.

9. Select the **Disable Language Dictionary to Correct Spellings** check box to disable the language dictionary spell corrector option for this field. If you have selected the **Use Language Dictionary to Correct Spellings** check box on the InScript2 Advanced Form Properties dialog, checking this option *disables* the language dictionary spell corrector for this field only.

10. Select the **Recognize Separately** check box in the Adjust Page-Level Settings area, to use a different language for this field. For example, if you want to process a form with English and French fields, these settings allow you to specify French as the language to use for the French fields, while all other fields "inherit" the English setting from the job or form-level settings.

11. In the **Language** field, select the to use for this field. See the Selecting a Language for InScript2 section for a list of the supported languages.

12. In the **Accuracy Bias** field, select from the following values. InScript2 applies the value to the language in the **Language** field. Options are:

    o **Accurate** – Uses the most accurate processing method, resulting in potentially more rejected characters.

    o **Fast** – Uses the fastest processing method. The processing bias is towards a lower reject rate, resulting in lower accuracy of non-rejected characters.

13. Select the **Configure** button to modify the settings for the sub-engine. Sub-engine configuration is supported at the field level for the following:

    o **Bar Code (Seaport 1- and 2-dimensional)**

    o Handprint (numeric only)

    o Handprint (RER reRecognition)

    See the Configuring InScript2 Sub-Engines section for more information.

14. Select **OK** to accept your changes and return to Designer.

15. Select **Save All** on the main Designer window to save your changes.

Configuring InScript2 Sub-Engines

You can set additional properties for the following sub-engines within InScript2:

- Bar Code (Seaport 1- and 2-dimensional)
- Handprint (numeric only)
- Handprint (RER reRecognition)

These settings are available at both the form and field levels.

Setting SEA (Bar Code) Sub-Engine Properties

The sub-engine properties for the Seaport bar code engine are set on the InScript2 Advanced Properties dialog. By changing the setting on this dialog, you fine tune the Seaport bar code engine for your type of bar code configuration. See the Understanding Bar Code Recognition section for an overview of the Seaport bar code recognition engine.

See the Creating a Bar Code Recognition Field section for general information and instructions on creating bar code fields on your form.

Follow these steps to set advanced properties for the SEA (Seaport 1- and 2-dimensional bar code) sub-engine.

At the form level:

1. Display the InScript2 Advanced Form Properties dialog. See the Setting InScript2 Form-Level Properties section.

2. In the **Sub-engine Configurations** field, select the Bar Code (Seaport 1,2 dimensional) option.

3. Select the **Configure** button. The InScript2 Bar Code Advanced Properties dialog displays.

4. Follow the steps in the At the InScript2 Bar Code Advanced Properties dialog section below.

At the field level:

1. Display the Field Properties, Recognition tab for the bar code field.

2. In the **Recognition type** field, select the **Bar code** option.

3. In the **Recognition engines** field, select the **InScript2** option.

4. Select the **Advanced Properties** button to display the InScript2 Advanced Field Properties dialog. See the Setting InScript2 Field-Level Properties section.

5. Select the **Recognize Separately** check box.

6. Select the **Configure** button. The InScript2 Bar Code Advanced Properties dialog displays.

7. Follow the steps in the At the InScript2 Bar Code Advanced Properties dialog section below.

At the InScript2 Bar Code Advanced Properties dialog:

1. In the **Barcode Type** field, select the bar code types you want to apply to your field or form.

2. In the **Orientation** field, select the general orientation for bar codes on the input image. For example, horizontal orientation indicates that the vertical bars of the bar code are

perpendicular to the x-axis of the image. Choosing a correct orientation option can improve recognition speed and accuracy. Options are:

- o **Horizontal only**
- o **Vertical only**
- o **Horizontal and vertical**
- o **Horizontal with skew**
- o **Vertical with skew**
- o **Horizontal and vertical with skew**

3. In the **Size (Inches)** area, set the general **Width** (physical size) of the bar code for this field or form. Width is defined as the direction perpendicular to the vertical bars of the bar code. The value does not need to be exact, but a close approximation can improve bar code recognition. Set a meaningful value in this field. If the bar code width is unknown, set this field to 0.

4. In the **Size (Inches)** area, set the general **Height** (physical size) of the bar code for this field or form. The value does not need to be exact, but a close approximation can improve bar code recognition. Set a meaningful value in this field. If the bar code height is unknown, set this field to 0.

5. In the **Quality** field, set a value from 1 to 10 with 10 being the best quality input image. Use this option to condition the bar code recognition process to assume different quality in images and thereby improve recognition accuracy and processing time. A lower quality value may decrease processing time, as the recognition engine will spend more time reading the bar code. A higher value will increase processing time. In general, use low values for poor quality images and high values for high quality images. Some examples for quality values are:

- o **FAX input – Quality = 1**
- o **First generation print – Quality = 10**
- o **2nd generation copy – Quality = 7**
- o **Microfilm scan – Quality = 4**

6. In the **Min chars** field, set the value to indicate the minimum number of characters that must be read in the bar code for the bar code to be considered valid. If fewer than this number of characters is found in the bar code, the bar code will not be reported in the count of valid bar codes found.

7. Select the **Fixed Barcode Length** check box to indicate that the bar code is of a fixed length. When this field is selected, each bar code must have exactly the number of characters indicated in the **Min chars** field to be valid. If this option is not selected, bar codes can be of variable length but must still pass the minimum length requirement in the **Min chars** field.

Caution: Use care when specifying this option. If you select both the **Enable checksum** option and the **Fixed Barcode Length** option, check digit(s) will *not* be counted as part of the fixed length bar codes. Likewise, when no checksum is performed, any check digits in the bar code data will be reported back as data; thus the specified length must include them.

8. Select the **Enable checksum** check box to indicate whether a checksum algorithm should be used to verify the bar code recognition results.

9. In the **INI override** field, select the **Browse** button adjacent to this field to locate an INI file to use *in place of* the settings you define on this dialog. The default location for INI files is the \FormWare\Ini\System folder.

10. Select **OK** to accept your changes on this dialog and exit.

11. Select **Cancel** to reject your changes on this dialog and exit.

12. Select **Reset** to reset all settings to their original state and exit.

13. Select **Save All** on the Designer window toolbar to save your changes.

Setting HNR (Handprint) Sub-Engine Properties

The sub-engine properties for the Handprint (numeric only) recognition engine are set on the HNR Sub-engine Properties dialog. By changing the setting on this dialog, you fine tune run the Handprint (numeric only) recognition engine for your type of handprint recognition. See the Understanding Handprint Recognition section for an overview of the Handprint (numeric only) recognition engine.

Follow these steps to set advanced properties for the Handprint (numeric only) sub-engine.

At the form level:

1. Display the InScript2 Advanced Form Properties dialog. See the Setting InScript2 Form-Level Properties section.

2. In the **Sub-engine Configurations** field, select the Handprint (numeric only) option.

3. Select the **Configure** button. The HNR Sub-engine Properties dialog displays.

4. Follow the steps in the At the HNR Sub-engine Properties dialog section below.

At the field level:

1. Display the Field Properties, Recognition tab for the handprint field.

2. In the **Recognition type** field, select the **Hand print** option.

3. In the **Recognition engines** field, select the **InScript2** option.

4. Select the **Advanced Properties** button to display the InScript2 Advanced Field Properties dialog. See the Setting InScript2 Field-Level Properties section.

5. In the Sub-Engine field, select the Handprint (numeric only) option

6. Select the **Recognize Separately** check box.

7. Select the **Configure** button. The HNR Sub-engine Properties dialog displays.

8. Follow the steps in the At the HNR Sub-engine Properties dialog section below.

At the HNR Sub-engine Properties dialog:

1. In the **Character Box Size** area, type the typical **Width** of the character box. The default value is 120. The combination of the **Width** and **Height** field values determines the dimensions (in 1/1200ths of an inch) of the typical character box for handprint characters on the form.

2. In the **Height** field, type the typical height of the character box. The default value is 0.

3. In the **Width of a space** field, type the typical width of a character space for handprint recognition. The default value is 0.

4. Select the **Output spaces found between digits** check box if you want to preserve the spaces found on the form in the output data file. The default value for this option is checked.

5. In the **Style of Handprint Digits** area, select **USA** or **EURO** to detect the slight differences in numeric handprint format between the United States and European countries. For example, Europeans use a hatch mark on the numeral 7; Americans do not.

Note: This setting affects handprint numeric characters only; it does not affect alpha characters.

6. Select **OK** to accept your changes on this dialog and exit.

7. Select **Cancel** to reject your changes on this dialog and exit.

8. Select **Reset** to reset all settings to their original state and exit.

9. Select **Save All** on the Designer window toolbar to save your changes.

Setting RER (Handprint) Sub-Engine Properties

The sub-engine properties for the Handprint (RER reRecognition) recognition engine are set on the InScript2 Handprint Advanced Properties dialog. By changing the setting on this dialog, you fine tune run the Handprint (RER reRecognition) recognition engine for your type of handprint recognition. See the Understanding Handprint Recognition section for an overview of the Handprint (RER reRecognition) recognition engine.

Follow these steps to set advanced properties for the Handprint (RER reRecognition) sub-engine.

At the form level:

1. Display the InScript2 Advanced Form Properties dialog. See the Setting InScript2 Form-Level Properties section.

2. In the Sub-engine Configurations field, select the Handprint (RER reRecognition) option.

3. Select the **Configure** button. The InScript2 Handprint Advanced Properties dialog displays.

4. Follow the steps in the At the InScript2 Handprint Advanced Properties dialog section below.

At the field level:

1. Display the Field Properties, Recognition tab for the handprint field.

2. In the **Recognition type** field, select the **Hand print** option.

3. In the **Recognition engines** field, select the **InScript2** option.

4. Select the **Advanced Properties** button to display the InScript2 Advanced Field Properties dialog. See the Setting InScript2 Field-Level Properties section.

5. In the Sub-Engine field, select the Handprint (RER reRecognition) option

6. Select the **Recognize Separately** check box.

7. Select the **Configure** button. The InScript2 Handprint Advanced Properties dialog displays.

8. Follow the steps in the At the InScript2 Handprint Advanced Properties dialog section below.

At the InScript2 Handprint Advanced Properties dialog:

1. In the **Noise Removal** area, select the **Enable Noise Removal** check box to enable noise removal prior to recognition processing. If you do not select this option, all of the controls in the **Noise Removal** area are disabled.

2. Select the **By Speckle Diameter** option to remove speckles (noise) by a specific number of pixels. If you select this option:
    o   Set the value in the **Speckle Diameter (Pixels)** field to the maximum diameter of pixels to remove.

3. Select the **By Stroke Percent** option to remove speckles (noise) by specific speckle length and width percentages. If you select this option:
   - o Set the value in the **Speckle Length (% of stroke)** field to the maximum stoke length dimension (as a percentage value).
   - o Set the value in the **Speckle Width (% of stroke)** field to the maximum stroke width dimension (as a percentage value).

4. Select the **No Edge Space Removal** check box to ignore (i.e., do not remove) the leading and trailing portions of surrounding white rows and columns during noise removal. During typical noise removal, isolated white spots on characters contaminated from outside the line area are deleted.

5. In the **Segmentation** area, define the segmentation of handprint characters. Select one or more of the following options:
   - o **Unconstrained** – By default all text is assumed to be constrained.
   - o **No Touching Characters** – By default the engine attempts to detect the touching characters and separate them. By checking this check box, this feature is being disabled.
   - o **No Overlapping Characters** – This feature works with the No Touching Characters option. If the No Touching Characters option is selected, then the No Overlapping Characters option is disabled. When the No Overlapping Characters option is deselected, the engine attempts to separate touching characters only along vertical lines.
   - o **Don't Use Alternatives** – When this feature is selected, the engine will attempt to locate alternative segmentation if the recognition results are unsatisfactory. Activating this feature may improve the recognition accuracy but it will degrade recognition speed because it is an "iterative" algorithm.

Note: If none of options in the **Segmentation** area are selected, the engine will attempt to automatically determine the optimal segmentation.

6. In the **CPI** area, define the general Characters Per Inch settings for handprint character recognition. Set the following options:
   - o **Min. CPI** – Defines the minimum CPI for the zone. The default is 0.
   - o **Max CPI** – Defines the maximum CPI for the zone. The default is 0.
   - o **Line height** – For multi-line fields, this value defines the line spacing. The default is 0, which means the engine automatically attempts to compute the line spacing.

7. In the **Positional Analysis** field, select from one of the following options:
   - o **Disabled** – Character is recognized regardless of its position in the line (raw results only). This is the default.
   - o **Medium** – The confidence of a character is adjusted to a limited degree based on its vertical position on the line.
   - o **High** - The vertical position of a character within the line strongly affects its recognition confidence.

8. In the **Context Analysis** field, select from one of the follow options:
   - o **Disabled** – No context analysis is used. This is the default.
   - o **Random** – Random distribution of character frequency is applied.
   - o **Native** – Language independent classifier is applied.

o **Language** – Language-specific information is used as specified in the job INI language parameter.

9. Select the **Fast Recognition** check box to increase handprint recognition speed but note that selecting this option may negatively affect handprint recognition quality.

10. In the **INI Override** field, select the **Browse** button adjacent to this field to locate an INI file to use in place of the settings you define on this dialog. The default location for INI files is the \FormWare\Ini\System folder.

11. If you choose to override the properties on this dialog with INI file setting, the **Unconstrained** text option (in the **Segmentation** area) and the **Min CPI** and **Max CPI** (in the **CPI** area) are automatically derived from the appropriate settings on the Field Properties, Recognition tab (similar to other recognition engines functionality). For example:

    o **If CPI is set to 0** – The setting will default to unconstrained.

    o **If CPI is not zero** – The text will be constrained, the minimum CPI will be set to 80% of the CPI value, and maximum CPI will be set to 120% of the CPI value.

12. Select **OK** to accept your changes on this dialog and exit.

13. Select **Cancel** to reject your changes on this dialog and exit.

14. Select **Reset** to reset all settings to their original state and exit.

15. Select **Save All** on the Designer window toolbar to save your changes.

Configuring the Mitek Recognition Engine

Use the Mitek recognition engine with FormWare to effectively recognize the following types of data:

- **Hand Print Text** – Mitek offers many choices for reading out-of-context alphanumeric handprint/handwritten text: uppercase only, upper- and lowercase, alphanumeric, and pure numeric. In addition, the system has capabilities for handling various punctuation marks frequently present on financial documents.

Note: Recognition of the punctuation characters in handprint text (activated by selecting the Preserve punctuation check box on the Field Properties, Recognition tab in Designer) requires a Mitek license that is enabled for CheckScript.

- **Machine Print Text** – Most machine print fonts can be recognized properly; this includes both fixed pitch and proportional fonts, in all common sizes.

- **Mark Sense Fields** – Mitek can test drop-out mark sense check boxes for content, i.e., whether a check mark or other marking has been placed in the box area.

- **Courtesy Amount Recognition (CAR)** – Mitek contains specialized routines that enable it to read a "courtesy amount" field on a check or other financial document; this field specifies the monetary amount using digits and frequently contains the currency symbol (e.g. "$") and possibly a fractional amount. When used with Mitek's add-on package, CheckScript, check processing capabilities are enhanced with CAR/LAR and CAR/LAR/MICR functionality. See Understanding Check Processing for more information.

Understanding Mitek and FormWare Integration

When used within a FormWare application, the Mitek engine supports the following:

Field Types

Mitek supports the following FormWare Field Types:

- Any Type
- Alpha Only
- Upper Case (Same as Any Type)
- Mostly Numeric (Same as Any Type)
- Unsigned Numeric
- Signed Numeric"

Characters Per Inch

The current implementation of the Mitek engine ignores the FormWare Characters Per Inch (CPI) setting.

OCR Font Types

Mitek supports the following OCR Font Types:

- REG
- OCRA
- OCRB
- MICR

Note: Farrington is not a supported font within Mitek.

Read Formats

Mitek supports the following FormWare Read Formats:

- Word
- Line
- Paragraph

Recognition Types

Mitek supports the following FormWare Recognition Types:

- Machine Print
- Hand Print
- Mark Sense
- Automatic
- CAR (For check processing applications. Only available with Mitek's CheckScript package)
- CAR/LAR (For check processing applications. Only available with Mitek's CheckScript package)

- CAR/LAR/MICR (For check processing applications. Only available with Mitek's CheckScript package)

Note: Bar Code and Cursive are not support recognition types within Mitek.

Segmentation

Mitek supports various kinds of segmentation, which can be selected via the Mitek Advanced Properties dialog. See Selecting a Segmentation Type for more information on Mitek segmentation.

Dictionaries

The Mitek engine does not currently support dictionary configuration for fields within FormWare. This may be added as an option in the future.

Free Form Recognition

The Mitek engine will support Free Form recognition within FormWare. During Free Form processing, each Free Form zone is passed separately to the engine. For more information on Free Form recognition, see Defining Free Form Recognition in Designing.

Field Point Size and Preserve Punctuation Options

Mitek supports both the field Point Size and Preserve Punctuation check boxes on the Field Properties, Recognition tab.

Field Include and Exclude Range Options

The Mitek engine does not support the Include and Exclude Range values on the Field Properties, Selective Edits tab. Refer to Defining Selective Edits for a Field in Designing for more information.

Field Input Masks

Mitek supports the field Input Mask functionality on the Field Properties, Selective Edits tab. Refer to Defining Selective Edits for a Field in Designing for more information.

Understanding Mitek Engine Requirements

Use of the Mitek recognition engine with FormWare requires the purchase and installation of both the Mitek software (as part of the FormWare installation) and a special Mitek hardware key that must be attached to the back of the workstation that will use Mitek recognition with FormWare.

If you are using Mitek recognition for a FormWare check processing application, you must purchase the Mitek hardware key with Mitek's add-on package, CheckScript, enabled. The CheckScript package enables recognition of CAR/LAR check data. The CAR/LAR recognition component is critical to successful check processing with the Mitek engine.

When you purchase a Mitek engine license to use with FormWare, you can choose from one of the following three options:

- Use Mitek QuickStrokes (without CheckScript) to process 25, 50, 100, 250, 500 and 1000 CPS (Characters Per Second) forms. Recognizes handprint text without

punctuation, machine print text, and MICR. No Courtesy Amount/Legal Amount recognition is available with this option.

- Use Mitek QuickStrokes with CheckScript to process 1, 2, 4, 6, 8, 10 and 12 Check PS forms. Recognizes Courtesy Amount, Legal Amount, and forms on personal checks. CheckScript combines Courtesy Amount Recognition technology (CAR) with natural handwriting recognition to provide reduced processing time and enhanced recognition accuracy for personal checks.

- Use Mitek QuickStrokes with CheckScript to process 1, 2, 4, 6, 8, 10 and 12 Check PS forms. Recognizes Courtesy Amount, Legal Amount and forms on personal and business checks. CheckScript combines Courtesy Amount Recognition (CAR) and Legal Amount Recognition (LAR) with natural handwriting recognition to provide reduced processing time and enhanced recognition accuracy for both personal and business checks.

Use the last two options for FormWare check processing applications.

Understanding Check Processing

To process checks using FormWare, you must use the Mitek QuickStrokes engine and its add-on package, CheckScript. CheckScript requires an additional license and special hardware key to run. The CheckScript-enabled key is different from the standard Mitek key but once installed, can be used for both regular Mitek recognition and check processing.

The QuickStrokes-enabled version of Mitek can effectively read the courtesy amount (CAR) on personal checks, business checks, and on other financial documents. The CAR amount is distinguished from other numeric fields in that there is a currency symbol (i.e., $) before the amount, and the fractional amount may be written using a numeric fraction, i.e., as a numerator with the denominator written as "00", "100", or as "XX".

CheckScript combines the Legal Amount Recognition (LAR) with the Courtesy Amount Recognition (CAR) to obtain better accuracy than a single method alone. The Courtesy Amount Read (CAR) is the numeric value on the check and is recognized first. The Legal Amount Read (LAR) field is the alphabetic data on the check and is recognized second to increase accuracy. The CAR and LAR values are compared for accuracy and run together when processing the check. When the results of the LAR value differ from the CAR, this can be useful in detecting fraud on the check. When LAR and CAR results are compared, the CAR amount has a higher weight, because the recognition is more reliable than on the LAR amount.

Note: When using the Mitek CheckScript package, you must scan all check images at 200 to 240 DPI.

Once installed as part of FormWare, Mitek CheckScript supports the following recognition types:

- **CAR** – Only recognizes the CAR value on the check and does not cross-verify this value against the LAR value. The CAR value is saved as OCR data during recognition processing.

- **CAR/LAR** – Recognizes both the CAR and LAR values on the check. The LAR value is compared to the CAR to verify its accuracy. Only the CAR value is saved as OCR data during recognition processing.

- **CAR/LAR/MICR** – Recognizes the CAR, LAR, and MICR (check routing information) values on the check. The LAR value is compared to the CAR to verify its accuracy. CAR and MICR values are saved as OCR data during recognition processing.

Processing Business Checks vs. Personal Checks

Business checks differ from personal checks in several important ways. Most business check are machine-printed, the CAR field location can vary, and the CAR amount may be preceded by filler characters, e.g. asterisks, which prevents someone from adding digits on the left-hand side. Select the proper CheckScript configuration for your type of check processing-business checks, personal checks, or both. See Understanding Mitek Engine Requirements for more information.

Note: CheckScript add-on can only be used to process U.S. personal and business checks. It is not meaningful to use it for other formats or non-dollar amounts.

When you set up FormWare to process check and read courtesy amounts (using a CAR recognition type), recognition success is based on Mitek license purchased. If you attempt to process business checks and are only licensed for personal checks, Mitek recognition processing will fail. When you use the CheckScript option to process business checks, the software verifies whether the check is a business check and, if it is, invokes the CheckScript business check logic. This includes special machine printed segmentation and support for the leading asterisks usually found in the CAR amount field of business checks. If you know that your application will only process personal checks, it is better not to use the business check configuration, since it could potentially slow your processing time in trying to determine the check type.

Creating a Check Processing Application

Note: These steps assume that you have purchased and installed the Mitek QuickStrokes recognition engine with the CheckScript add-on package as part of your FormWare installation and have attached the Mitek hardware key with the appropriate licensing to the back of your workstation.

Follow these steps to create a FormWare check processing application that uses the Mitek recognition engine:

1. Create a check-processing job with at least one form within the Designer module.

2. Display the **Form Properties > Image** tab, select a check image as the setup image for this form, and run the setup image through IPP processing. See Selecting a Setup Image in Designing for more information.

3. Create a field on the data form to collect the CAR data. Make sure that the field is large enough to hold the full numeric amount including any punctuation. Add a label for this field if you wish.

4. With the CAR field selected on the data form, draw a recognition zone of any size anywhere on the setup image.
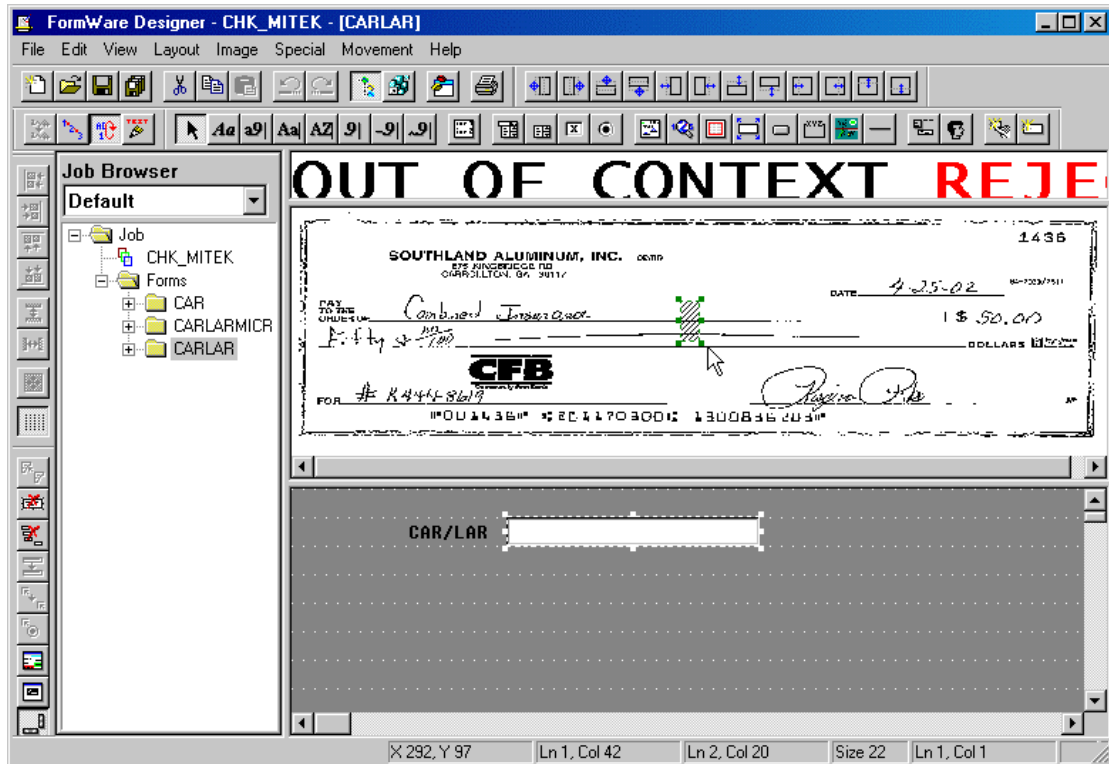
5. With the field selected on data form, display the **Field Properties > Recognition** tab. Set the following field properties:

    a. Set the Recognition type to CAR/LAR.

    b. Set the Recognition engine to Mitek.

    c. Optionally, check the **Preserve Punctuation** field to retain field punctuation.

6. Select **Test** to test field-level recognition. This launches the Mitek recognition engine and results display within the field on the data form. Detailed OCR results also display beneath the Job Browser. See Testing Recognition in Designing for more information on field OCR results.
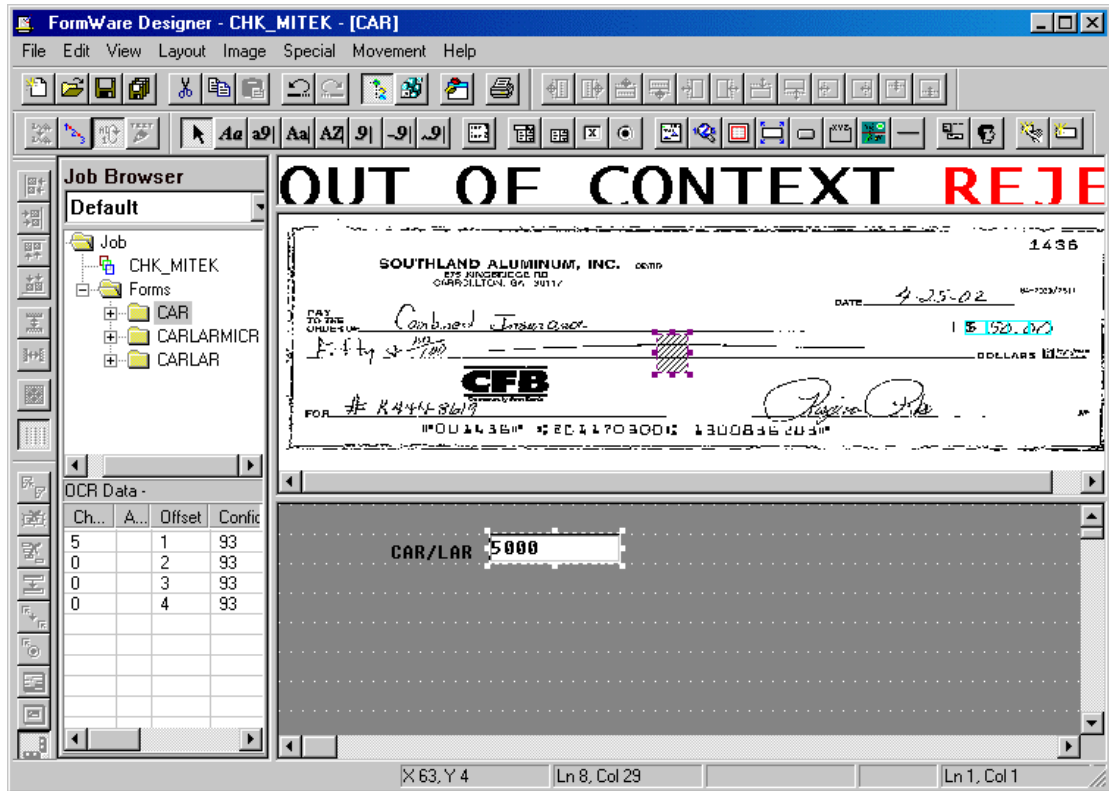
7. When the CAR field is successfully recognized, blue segmentation boxes appear around the field characters on the setup image.

8. Select **Save All** on the Designer window to save your work.

Capturing Check Routing Information

Check routing information appears at the bottom of check and identifies the banking institution and account information associated with the check. In addition to the CAR value, check routing information can be captured and stored as OCR data.

For the purposes of FormWare check routing information is called the MICR data because that is the font standard for this type of data on checks. When you choose to capture the MICR line, you are also capturing a number of special symbols that are unique to check routing. These symbols delineate the start parts of the check routing line. For the purposes of FormWare OCR, each of these symbols has been mapped to an alpha character to optimize recognition performance: The mapped symbols are as follows:

MICR Symbol	Description	OCR Output
||*	Customer Account Number	c
.|*	Amount of Check	b
·|:	Bank Branch Number	d
–	Dash	–

Follow these steps to capture check routing information within a FormWare check processing application:

1. Follow the steps in Creating a Check Processing Application to install and set up a check-processing job.

2. Create a field on the data form to collect the MICR data. Make sure that the field is large enough to hold the full numeric amount including all symbols within the MICR line. Add a label for this field if you wish.

3. With the MICR field selected on the data form, draw a small recognition zone in the center of the setup image.

Note: The zone does not need to be on the actual MICR line of the setup image.

4. With the field selected on data form, display the **Field Properties** > **Recognition** tab. Set the following field properties:

   a. Set the Recognition type to CAR/LAR/MICR.

   b. Set the Recognition engine to Mitek.

   c. Set the Read format to Paragraph.

   d. Optionally, check the Preserve Punctuation field to retain field punctuation.

5. Select **Test** to test field-level recognition. This launches the Mitek recognition engine and results display within the field on the data form. Detailed OCR results also display beneath the Job Browser. See Testing Recognition in Designing for more information on field OCR results.

6. Select **Save All** on the Designer window to save your work.

Understanding Mitek Engine Advanced Field Properties

You can also fine-tune Mitek engine advanced properties for individual fields to improve:

- **Segmentation** – Segmentation is a method used to extract the individual characters from the original image before passing them on to the recognition step. There are many different segmentation methods. Deciding which is the best one to use depends on: the format of the text in a field (segmentation is chosen on a field-by-field basis), constraints imposed on characters, and the level of noise present. See Selecting a Segmentation Type for more information.

- **Line Removal** – Sometimes, fields have lines running around or even through them. Such lines can be removed without disturbing the underlying text, as long as the lines are horizontal or vertical and have a certain minimal (unbroken) length. The minimum length threshold can be adjusted, but it is obviously bounded by the size of the characters in the field: if line segments are not much longer than horizontal or vertical parts of characters (e.g. the digit 1), then such parts would be removed by mistake, causing recognition errors. Use Mitek's advanced line removal functionality to effectively remove such extraneous items as boxes, combs, and other lines on the form.

Selecting a Classifier

Within Mitek, a wide variety of neural classifiers are available for both machine print and handprint fields. Classifiers enable you to designate a subset of characters and symbols for a particular field, thereby optimizing the recognition performance on that field's data. When you select a particular classifier for a field on the Mitek Advanced Properties dialog, the character set

supported for the classifier is displayed for your information. Classifiers for machine print text, handprint text, and special fonts like OCRA and MICR are not hardwired into the Mitek engine but are retrieved by the engine from the \FormWare\Mitek\Sys directory. This allows Mitek to update the classifier set or provides custom classifiers with a minimum amount of disruption.

Selecting a Segmentation Type

Next to actual recognition, segmentation is probably the most important part of performing text reading. Segmentation is the method of extracting each individual character from the field. The different methods available are split into three main categories: for hand-printed text, for machine-printed text, and special-purpose methods (i.e., check processing). Some methods are useful for more than one category.

When you define advanced segmentation settings for a field processed by Mitek, you specify a segmentation option, which determines how the individual characters are extracted from it. In the simplest case, namely "isolated characters" or "semi-constrained characters", your application determines where each character will be located, and all the recognition software has to do is extract the bitmaps from the areas specified. In many cases, however, the choice of a segmentation option depends on the data present in that particular field. More detail of some of the more common types of segmentation follows.

Semi-Constrained Segmentation

This segmentation combines the speed of isolated segmentation, which saves time by looking for characters in predefined locations, with some of the flexibility of unconstrained segmentation, which is able to find characters whose locations are not known in advance. In most cases this segmentation supersedes isolated segmentation, incurring only a minor speed penalty.

Semi-constrained segmentation is best used in cases where the actual data (characters) are not written exactly into their respective boxes, even though such boxes are defined (just like for isolated text). In other words, the writers of the text were not very careful in placing the text. This often happens when the character boxes are directly adjacent, i.e., when there is no space between the boxes, either horizontally or vertically or both. Actually, this segmentation type was originally developed for fields with no space vertically between lines nor horizontally between characters.

The segmentation algorithm will search outside of the original character area (in all four directions) if the text seems to go beyond it (i.e., if it touches the area borders), and will thus expand the original box if necessary. As a rule of thumb, the expansion will not be more than one half the size of the original area in any one direction. If your application does not require the fastest speed possible, then using semi-constrained segmentation instead of isolated segmentation will very often yield better recognition results at only a small performance penalty.

Isolated and Semi-Constrained Handprint Segmentation

Semi-constrained and Isolated segmentation for handprint test requires that you manually set the character coordinates for each character in the field. Use the Character Boxes tool to define the individual character boxes for proper field segmentation. Access this dialog via the Character Positions button on the Mitek Advanced Properties dialog for handprint fields. See Setting Advanced Properties for Handprint Fields for instructions.

Machine Font Segmentation

Machine Font segmentation differs from other types of segmentation in that it takes into account the information collected from other fields (all fields in the same mask using the same

segmentation). Since the algorithm collects information across multiple fields, the following rules apply:

- All fields must contain text printed using the same font, of the same size. However, all machine fonts and sizes are allowed (just not at the same time).

- The text must contain straight separate lines; note that skewed lines are acceptable, as long as they still have a straight base line. Touching lines will impair performance. Long, closely spaced lines have only marginal skew tolerance, while short and/or widely spaced lines can tolerate greater skew.

- The more characters are found with this segmentation (and can therefore be used to determine the font characteristics), the better the results will tend to be.

Note: If your form contains fields printed with various fonts or sizes and you want to use this segmentation on all those fields, you must define a separate mask for each homogeneous group of fields, and then perform recognition separately on each mask. This will prevent the segmentation processing from mixing information about the different kinds of fields.

Setting Advanced Field Properties for Mitek

When you use the Mitek engine to recognize handprint, machine print, or automatic text, you can set additional parameters at the field level that allow you to fine-tune Mitek recognition on that field. The Mitek Advanced Properties dialog is activated according to the recognition type you select on the Field Properties, Recognition tab:

- When the recognition type for the field is defined as Hand Print, all the Machine Print settings are disabled.

- When the recognition type for the field is defined as Machine Print, all Hand Print settings are disabled.

- When the recognition type for the field is defined as Automatic, both Hand Print and Machine Print settings are enabled.

Setting Advanced Properties for Handprint Fields

Note: These steps assume that you have purchased and installed the Mitek QuickStrokes recognition engine as part of your FormWare installation and have attached the Mitek hardware key with the appropriate licensing to the back of your workstation.

Follow these steps to set Mitek advanced properties for handprint fields:

1. In Designer, create a job with at least one form.

2. Display the Form Properties, Image tab, select a check image as the setup image for this form, and run the Image through IPP processing. See Selecting a Setup Image in Designing for more information.

3. Create a field on the data form to collect the handprint data. Make sure that the field is large enough to hold the maximum number of characters the data could potentially be including any punctuation. Add a label for this field if you wish.

4. With the field selected on the data form, draw a recognition zone over the area on the setup image representing the entire field.

5. Display the Field Properties, Recognition tab. Set the following field properties:

   a. Set the **Recognition type** to Hand Print.

b. Set the Recognition engines to Mitek.

c. Optionally, set the remaining fields on this dialog or leave at their defaults.

6. Select the **Advanced Properties** button to display the Mitek Advanced Properties, General tab.

Note: Only the Hand Print Settings section of this page is active for hand print fields.

---

7. In the **Classifiers** field, select the option that best determines the type of data that will be in this field. Once you select a classifier value, the Character Set field displays a representation of the character set within that classifier.

8. In the **Segmentation** field, select the appropriate value that represents the type of character segmentation in this field. Valid values are:

  o **Isolated** – Best for handprint fields where each character's position is known in advance and you do not expect the writer to violate the character box boundaries.

  o **Semi-Constrained** – Use for handprint field characters for that have individual boxes defined but where you expect some characters to protrude outside of those boundaries.

  o **Unconstrained** – Use for handprint fields where locations of characters are unknown; works on single-line fields only.

  o **Blobs** – Use for single-line handprint fields where program will automatically insert breaks between words. Similar to Unconstrained segmentation for handprint fields.

  o **Groups** – (Page Reader Option) Use for multi-line handprint fields where program will automatically insert breaks between words. This is the only segmentation for multi-line hand printed text.

9. If you selected Isolated or Semi-Constrained in the **Segmentation** field, the **Character Positions** button activates. Select it to display the Character Boxes dialog.

10. Use the mouse to draw boxes around each character in the field.

11. Activate an existing box by clicking on it with the mouse. An activated box displays black handles around it. You can grab the entire box to move it to a new location or a box handle to stretch or shrink the box.
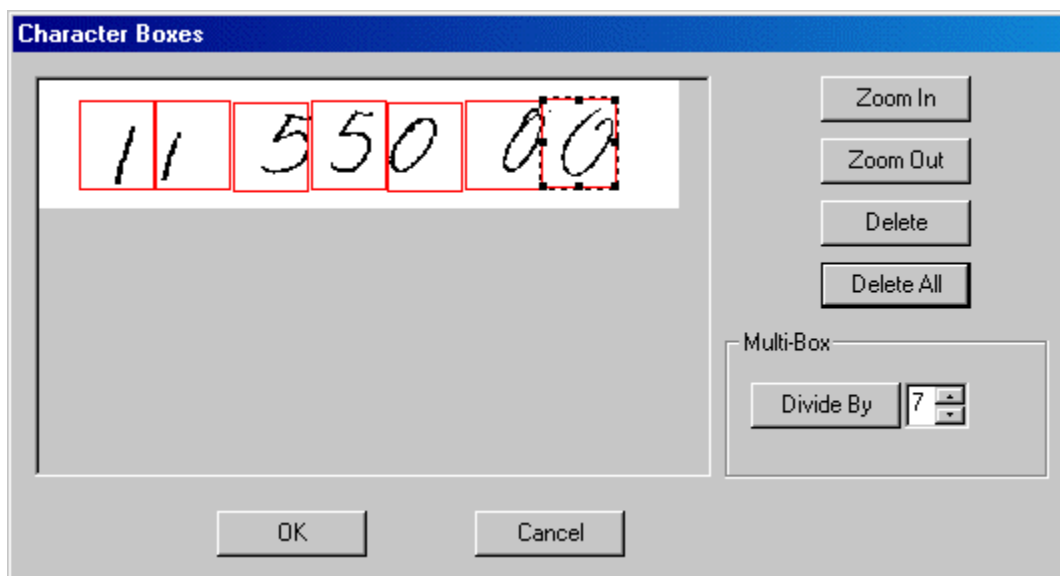
12. Delete a single active box using the **Delete** button.

13. Delete all boxes on the field using the **Delete All** button.

14. Use the **Zoom In** and **Zoom Out** buttons to enlarge or shrink the field.

15. To use the **Multi-Box** option:

    a.   Activate an existing box.

    b.   Change the Multi-Box number to the number of boxes you want to create.

    c.   Press the **Divide By** button.

16. Select **OK** to accept your changes or **Cancel** to abort your changes and return to the Mitek Advanced Properties dialog.

17. Select the **Segmentation Properties** tab to set advanced segmentation properties for this handprint field.

Caution: The settings on this tab are intended for advanced users only. Refer to the Mitek QuickStrokes documentation for more information on the settings on this dialog and how they impact field recognition.

18. Select the **Zone Processing Properties** tab to set advanced segmentation properties for this handprint field.

Caution: The settings on this tab are intended for advanced users only. Refer to the Mitek QuickStrokes documentation for more information on the settings on this dialog and how they impact field recognition.

19. Select **OK** to accept your changes and return to the **Field Properties, Recognition** tab.

20. Select **Test** to analyze your field recognition settings. This launches the Mitek recognition engine. Results display in the field on the data form.

21. Once all test results are successful, select **Save All** on the Designer window to save your changes.

Setting Advanced Properties for Machine Print Fields

Note: These steps assume that you have purchased and installed the Mitek QuickStrokes recognition engine as part of your FormWare installation and have attached the Mitek hardware key with the appropriate licensing to the back of your workstation.

Follow these steps to set Mitek advanced properties for machine print fields:

1. In Designer, create a job with at least one form.

2. Display the Form Properties, Image tab, select a check image as the setup image for this form, and run the image through IPP processing. See Selecting a Setup Image in Designing for more information.

3. Create a field on the data form to collect the machine print data. Make sure that the field is large enough to hold the maximum number of characters the data could potentially be including any punctuation. Add a label for this field if you wish.

4. With the field selected on the data form, draw a recognition zone over the area on the setup image representing the entire field.

5. Display the Field Properties, Recognition tab. Set the following field properties:

a. Set the **Recognition type** to Machine Print.

b. Set the Recognition engines to Mitek.

c. Optionally, set the remaining fields on this dialog or leave at their defaults.

6. Select the **Advanced Properties** button to display the Mitek Advanced Properties, General tab.

Note: Only the Machine Print Settings section of this page is active for machine print fields.

---

7. In the **Classifiers** field, select the option that best determines the type of data that will be in this field. Once you select a classifier value, the **Character Set** field displays a representation of the character set within that classifier.

8. In the **Segmentation** field, select the appropriate value that represents the type of character segmentation for this machine print field. Valid values are:

    o **Unconstrained** – Use on machine print fields when locations of characters are unknown; works on multi-line and proportional texts.

    o **Monospaced** – Use for machine print fields made up of monospaced characters; works on multiple lines and character columns need not be aligned; marginal tolerance for field skew (rotation).

    o **Monospaced Label** – Use for machine print fields made up of monospaced characters; if field contains several lines, the columns must be aligned and the vertical line spacing must be constant; for fields matching this description, this option provides superior recognition performance with good tolerance for field skew (rotation).

    o **Machine Font** – Use for all machine print fields on a form containing the same font (of the same size). Works on multi-line fields with straight, separated lines of text.

    o **Machine Field** – Same functionality as Machine Font, but works better than Machine Font on a single field.

    o **Grotesque** – Use for special stylized machine print fonts, such as CMC7. The characters in these fonts are composed of single isolated vertical or horizontal bars. These fonts are a combination of bar codes and regular fonts such as OCR-B.

9. In the **Rotation** field, select the option that determines this field's orientation on the data form. For example, when recognition is processing a field printed along the vertical border of the page (at 90 degrees rotation). Rotation settings are calculated in a clockwise direction. Options are:

    o **None** – No rotation (default)

    o **90 Degrees**

    o **180 Degrees**

    o **270 Degrees**

10. Select the **Segmentation Properties** tab to set advanced segmentation properties for this handprint field. The settings on this tab are identical to those set for handprint fields.

Caution: The settings on this tab are intended for advanced users only. Refer to the Mitek QuickStrokes documentation for more information on these settings and how they impact field recognition.

---

11. Select the **Zone Processing Properties** tab to set advanced segmentation properties for this handprint field. The settings on this tab are identical to those set for handprint fields.

Caution: The settings on this tab are intended for advanced users only. Refer to the Mitek QuickStrokes documentation for more information on these settings and how they impact field recognition.

12. Select **OK** to accept your changes and return to the Field Properties, Recognition tab.

13. Select **Test** to analyze your field recognition settings. This launches the Mitek recognition engine. Results display in the field on the data form.

14. Once all test results are successful, select **Save All** on the Designer window to save your changes.

Understanding the Mitek.ini File

The Mitek.ini file contains the following sections and parameters:

[Diagnostics] Section

Caution: The parameters within this section are for internal troubleshooting purposes only and not intended to be left on within a production environment. Contact Captiva's Technical Support department for further information.

Enable log = yes/no

When set to yes, this parameter enables the log file generation by the engine. The log file is generated to the operating system's \TEMP directory. The name of the log file is MitekDiag.txt. The contents of the log file are overwritten each time the engine starts. The default is setting is No.

Log Level =

This parameter works with the Enable log parameter and defines the level of detail that is created within the log file. Unless specifically requested by Mitek, this value should always be set to 3 (error only logging). The default is setting is 3.

Save FRM = yes/no

Set this parameter to "yes" to save the Mitek form descriptor to the operating system's \TEMP directory for further analysis. The name of the file is <FDF filename>mtk.frm (For example 1040ezmtk.frm). The default is setting is "no."

Disable Extra Spaces = yes/no

Set this parameter to "yes" to disable the space extrapolation algorithm. Because the Mitek engine always places only one space between the words for unconstrained segmentation (regardless of the gap size between the words), Captiva's space extrapolation algorithm inserts extra spaces when needed. This setting disables the space extrapolation algorithm. The default is setting is "no."

Non Fatal Codes = -20,-21

Normally all the errors returned by the recognition engine are treated as fatal errors and cause batch processing to stop. However, errors that only apply to a single image should not terminate the entire batch. By inserting the error numbers of non-fatal errors in a comma-delimited list for

the Non Fatal Codes parameter, you can omit the specified errors from terminating batch processing. There are currently two valid error codes you can insert in this parameter:

- **-20** – Check is too noisy
- **-21** – Check has too many broken characters.

A single 0 in the non fatal codes list means all the codes returned by the engine are treated as not fatal and do not cause batch termination. The default value is "-20,-21."

[Defaults] Section

The parameters in this section allow you to specify the default values that appear for certain Mitek engine advanced properties. There are two parameters:

```
def1 =#,#,#,#,#,#,#
def2 =#,#,#,#,#,#,#
```

The format of the parameters are as follows:

```
defnn= type,font,classifier,flag,parm,intvalue,fltvalue
```

Valid values are:

- *nn* – The index of the parameter (start with 1 and increment without any gaps);
- *type* – The recognition type for the zone (0 is any, 1 is machine print, 2 is hand print 3 is automatic, and 4 is cursive)
- *font* – The font flag (0 – applies for to fonts, 1 – applies to REGULAR, 2 – applies to OCRA, 3 – applies to OCRB, 4 – applies to MICR, 5 – applies to FARRINGTON)
- *classifier* – The classifier to which the setting applies (0 means any classifier)
- *flag* – The mode flag is 1 (for MODE1) or 2 (for MODE2)
- *parm* – The parameter number from the Mitek manual
- *intvalue* – The integer value of the parameter
- *fltvalue* – The float value of the parameter

For example the setting def1= 1,4,0,2,4007,35,0.0 indicates that the setting applies to machine print (1), MICR font (4), any classifier (0), it is MODE2 (2), the parameter number is 4007 (CHAR_BREAK_MAX (minimize wide characters) for MICR recognition) the Mitek parameter is set to 35.

When any of a form's fields use the same advanced setting as set in Mitek.ini file, but are set via that field's advanced properties dialog, the advanced properties dialog settings always overrides the default values set within the Mitek.ini file.

[Classifiers] Section

The entries in this section establish the default association between the Captiva's field (recognition) types within Designer and the Mitek classifiers used to recognize these types of fields. This default can always be overwritten on the Mitek Advanced Properties dialog.

[CAR/LAR] Section

The entries in this section define the names of the Mitek form descriptors to be used for recognition of CAR, CAR/LAR and CAR/LAR/MICR fields.

Note: Although though the names of the .frm files can be overwritten, the location of those files cannot. The form descriptors are always located in \FormWare\Mitek directory.

Generating Engine Error Diagnostics

When problems occur with the Mitek engine, e.g., an engine crash, engine error, or unsatisfactory recognition results, Mitek can generate an error log file to help you troubleshoot the problem. The combination of this error log and the Mitek-specific form descriptor can provide vital information to solve the issue or enable further analysis.

Generate the error log file by setting the Enable log = parameter to Y (Yes) in the Mitek.ini file. This generates the event log file to the Windows temporary directory. The name of the log file is MitekDiag.txt. In addition the Mitek's form descriptor will be saved in the same directory. The name of the form descriptor is <FormName> MTK.FRM (for example 1040ezMTK.FRM). These two files can be sent to Captiva's technical support for analysis. See Understanding the Mitek.ini File for more information on diagnostic settings.

Defining Recognition Zones

When you create a recognition zone for a field, you must set the recognition properties for that field. You set the default recognition properties on the Job Defaults dialog when you first created the job.

Setting Recognition Properties for a Field

To change these default recognition properties perform the following steps:

1. Open a job and form in Designer.

2. Move the image within the Image Viewport so the desired area on the setup image for the field displays in the Image Viewport.

3. Create a field on the data form.

4. On the Image Viewport, select a corner of the area on the image and drag the cursor to cover the area, and then release the mouse. The recognition zone displays over the area you just marked. If the recognition zone contains any form background that requires removal, configure removal features on the Field Properties, Enhancement page prior to testing recognition.

Note: The recognition zone remains active and vulnerable to any mouse movements you make until you deselect the associated field on the data form.

5. Display the Field Properties, Recognition tab.

6. In the **Confidence** field, select the value that specifies how confident the recognition engine needs to be to accept characters. All characters with confidence levels below this value appear as character rejects during a Completion session.

7. In the **Characters Per Inch** field, select the number of characters per inch on the original document. This is available with the Nestor, AEG, and CGK engines. Setting this option, which helps the recognition engine segment the characters in the field correctly, can improve recognition. To determine the CPI, measure a one-inch segment on an original form and determine how many characters are located in the inch. For example, if your recognition zone is two inches long and your CPI is 5, the recognition engine would slice the zone into ten equal sections and then try to recognize a character in each zone. This setting applies to all the fields on your form.

8. You can also select particular fields on the Data Form and change the CPI setting for the selected fields only. Click and drag your mouse to highlight the fields you want to change. With the selected fields highlighted, right-click to display the Context menu. Select the **Select Chars Per Inch** option. Change the setting in the CPI dialog and click **OK** to continue. This setting change applies only to the highlighted fields.

9. In the **Errors Allowed** field, select the number of low-confidence characters allowed in a field before the field is rejected during Completion. When the number of error characters in a field exceeds this value, the first character is replaced by the OCR correction character (~) and the remaining characters will be spaces (i.e., the data is preserved, but the field displays spaces). For example, if you select 5 in this field and 7 characters are rejected, the whole field is rejected. If 4 characters are rejected, the individual reject characters are displayed in Completion.

10. In the **Data Offset** field, select the position in the recognition results file to begin reading data for this field.

11. In the **Data Length** field, select the number of characters to read into this field from the recognition results file.

12. In the **Point Size** field, select the minimum point size of characters. Any smaller characters are ignored.

13. Select the **Preserve Punctuation** check box to indicates whether punctuation (e.g., question marks, periods, and hyphens) is included in output data. If you do not select this option, these characters are deleted.

14. In the **OCR Font** field, select a trained font to use on the field. The Nestor Reader engine supports OCRA, OCRB, Farrington, and MICR. You must purchase the Nestor trained fonts separately and they can only be used on numeric fields. The AEG engine supports OCRA for alpha and numeric data, OCRB and the Farrington fonts for numeric data, and MICR.

15. In the **Read Format** field, select the value that determines how data in a field is treated by Recognition. Options are:

    o **Word** – Do not allow spaces in the data field. If spaces are encountered, they are compressed out of the data string for the field.

    o **Line** – Allow spaces in the data field. If spaces are encountered, they are preserved in the data string for the field.

    o **Paragraph** – Extracts data from multiple lines on the image with a single recognition zone. Lines within the paragraph are auto-detected and parsed into the appropriate data field.

16. In the **Recognition Type** field, select the type of data for which the recognition engine will search in the recognition zone for the field. Options are:

    o **None** - No recognition is performed for the current field. This type does *not* allow for advanced settings when using the AEG, CGK, or RecoStar recognition engines. This is the default setting.

    o **Machine Print** - Recognizes machine-printed alphanumeric characters that have consistent, predictable shapes. This type allows for advanced settings when using the AEG, CGK, or RecoStar recognition engines. Use this option to apply free form recognition to a field.

    o **Hand Print** - Recognizes alphanumeric characters that vary in shape, such as hand-printed characters. This type allows for advanced settings when using the AEG, CGK, or RecoStar recognition engines.

- o **Mark Sense** - Recognizes the checkmarks, Xs, or other marks placed in check boxes. This type does *not* allow for advanced settings when using the AEG, CGK, or RecoStar recognition engines.
- o **Bar Code** - Recognizes industry-standard bar codes. Bar codes are self-contained information that is encoded in the widths of printed bars and spaces. This type does *not* allow for advanced settings when using the AEG, CGK, or RecoStar recognition engines.
- o **Automatic** - Attempts to automatically determine the type of characters to be recognized. This type allows for advanced settings when using the AEG, CGK, or RecoStar recognition engines.
- o **Cursive** - Recognizes cursive handwritten characters. This type allows for advanced settings when using the AEG, CGK, or RecoStar recognition engines.

Note: Not all recognition engines support all recognition types. See the Understanding Recognition Types and Properties section for more information.

17. In the **Segmentation** field, select the value that provides the recognition engine with an estimate of how the characters are spaced, providing a means of determining where each character should begin and end. Segmentation is the amount of spacing between characters. If you have characters that are very wide, set the segmentation to 5. If you have characters that are close together, set the segmentation to 1. If you specify None, the engine attempts to determine the segmentation size.

Note: The **Segmentation** field is used by the Nestor engine. None of the current engines use it. RecoStar, InScript2, and Mitek have advanced property settings that affect the segmentation. The **Characters per inch** (CPI) value sets the segmentation, but only if this is not overridden by an advanced property setting. Typically, a setting of "0" forces machine print to "proportional" spacing, and handprint to "unconstrained". A non-zero value causes "constrained" handprint segmentation at that CPI. In InScript (but not InScript2), if all of the fields on the form are set to the same, non-zero CPI, then recognition is done in "fixed" pitch mode, rather than proportional, for the whole page. If some fields are "0" and some are "fixed", then the whole page is processed in "automatic" mode.

18. In the **Recognition Engines** field, select the recognition engine(s) to use for this field. If you have data from a recognition engine that is not listed, but is FormWare compliant, select the **Other** option. You can specify multiple engines to be executed on a field. However, you need to enable recognition voting if you are using multiple engines. See the Selecting a Recognition Engine section for more information.

19. In the **Read Length** area, select the option that determines how the recognition engine will read characters in this field. Options are:
    - o **Fixed** – Indicates that the number of characters read by the recognition engine is expected to equal the length of the field. Field data that does not equal the field length is rejected.
    - o **Variable** – Indicates that the number of characters read by the recognition engine does not need to equal the length of the field. The data can be any length less than or equal to the field length.

20. Select **Save All** on the Designer window toolbar to save your changes before you exit the Designer module.

Working with Recognition Zones

You can adjust the placement of a recognition zone as well as copy and delete recognition zones.

Adjusting Zone Size and Placement

Follow these steps to resize or move a recognition zone on the setup image.

1. Activate a zone on the image by clicking on the associated data field on the data form.

2. Move your cursor over the zone in the Image Viewport. When the bi-directional arrow displays, click and hold your mouse and move the edge of the zone until it is the appropriate size.

3. To move the zone to a new location, move the cursor over the zone until the quad-directional arrow displays. Click and drag the entire zone to the correct location.

4. To move the image, move the cursor over a non-zoned area of the setup image and right-click and drag the image to the correct position. The zone stays in the same location and the image moves with the mouse.

5. Select the **Save All** button on the toolbar to save your changes.

Copying Zones

To copy a zone and its viewport origin from one field to one or more fields:

1. Click the field on the data form from which the zone is to be copied. While the field is highlighted, press the [Shift] key and click the field(s) to which you want to copy the recognition zone.

2. Release the mouse and select the **Copy Zone Info** button. The zone is copied from the first field selected to all other selected fields.

3. Select the **Save All** button to save your changes.

Deleting Zones

To delete a zone that is associated with a field:

1. Activate a zone by clicking on the associated data field on the data form.

2. Select the **Delete Zone** button on the Image toolbar. To delete all of the zones on a form, select the **Delete All Zones** button. When selected, the **Delete All Zones** function prompts you before deleting all zones.

3. Select the **Save All** button on the Designer window toolbar to save your changes.

Creating a Bar Code Recognition Field

Bar code recognition can provide fast, reliable results if used correctly. Understanding the general structure of a bar code and how to correctly identify the bar code type are essential to success. Bar codes contain the following basic elements:

- **Quiet zone** - The quiet zone occurs at the beginning and end of bar codes. No information is coded in these areas.

- **Character sequence** - The character sequence contains the data to be captured.

- **Start pattern** - The start pattern is the sequence of bars after which data is present. The start pattern signifies the beginning of a bar code.

- **Stop pattern** - The stop pattern is the sequence of bars indicating the end of valid data.
- **Check digit(s)** - Certain bar codes require check digits as the last character. Some bar codes support more than one check digit.

When you create a recognition zone on a bar code, you typically do not need to cover the entire bar code. Instead, it is best to create a zone that covers a strip of the bar code and includes the quiet zone, or white space on both sides of the bar code.

Caution: If you use bar code recognition with Image Enhancement and have selected the **Remove Background** option, you can experience problems with bar code recognition. When the image is enhanced and the background is removed, only the portion of the image inside the recognition zone is retained, eliminating part of the bar code. Bar code recognition generally fails to produce accurate results under these circumstances. If you must remove the background to process other fields correctly, then you must completely enclose the bar code within the recognition zone. Include as much white space to the left and right of the bar code as possible.

Understanding Bar Code Types

FormWare supports the following bar code types. Successful recognition depends on correct identification of the bar code. The characteristics of the supported bar code types are as follows:

- **Codabar** - Binary bar code of variable length. The valid character set includes 0-9 and the special characters "$", "-", ":", "/", ".", and "+". A series of wide and narrow elements are used for character representation. No check digit is required with use of the Codabar.
- **Code 128** - Proportional bar code of variable length. The valid character set includes all 128 ASCII characters. This bar code, comprising six elements (three bars and three spaces), is very dense and compact. With the exception of the stop character, all other characters are 11 units long.
- **Code 39** - Binary bar code of variable length. The valid character set includes A-Z, 0-9, and the special characters "$", "-", ":", "/", ".", and "+". The asterisk (*) is used as the start and stop character.
- **Code 39 Extended** - Extended version of Code 39. It is exactly like Code 39, but the full ASCII character set is available.
- **Code 2 of 5** - Numeric-only binary bar code of fixed length. It uses five bars and five spaces per character.
- **Code 2 of 5 Interleaved (Code 2 of 5I)** - Exactly like Code 2 of 5, except that white spaces also represent a character. In Code 2 of 5, white spaces are only used as separators between two bars.
- **UPC-A (Universal Product Code)** - Numeric-only, proportional bar code with a fixed length of five (four characters and one check digit).
- **UPC-E** - Numeric-only, proportional bar code with a fixed length of six (five characters and one check digit).
- **EAN-8 (European Article Numbering)** - Numeric-only, proportional bar code that encodes eight digits. There are seven characters and one check digit.
- **EAN-13** - Numeric-only, proportional bar code that encodes 13 digits. There are 12 characters and one check digit.
- **PostNet (Postal Numeric Encoding Techniques)** - Numeric-only, binary bar code. The length is 5, 9, or 11 digits. All variations have one check digit.

- **Code 93** - Proportional bar code of variable length. The valid character set includes A-Z, 0-9, the space, four shift characters, and the special characters "$", "-", "%", ".", "+", and "/". This structurally complex bar code requires two check digits.

Understanding Image DPI and Bar Code Type

Verify that the DPI of your images is adequate for the size and type of bar code you use. The per character pixel requirements for the following bar codes are:

- Code 3 of 9 - 50 pixels per character
- Code 2 of 5 - 32 pixels per character
- Codabar - 32 pixels per character
- Code 128 - 44 pixels per character

Bar codes are usually easy to read and reject rates are low; however, a few factors can impact performance. Resolution is the main factor. We recommend limiting the number of characters per bar code inch to 4 at 200 DPI and 6 at 300 DPI.

Keep in mind that the height of a bar code should not be less than 25% of its width. Print and paper quality can also affect the performance of a bar code read. Laser printers and preprinted bar codes offer the best contrast and cleanest separation between bars. Dot matrix printers are affected by the quality of ribbon.

Bar codes are typically simple codes that can be decoded very quickly. With Visionshape Software Solutions, a bar code can usually be read in less than 1/4 second on a 486/66 MHZ PC.

Defining Recognition for a Bar Code Field

To create a bar code field:

1. Add an Any Type field to your form and draw a zone around the bar code on the image.
2. Double-click the field to display the Field Properties dialog for that field.
3. Select the Recognition tab.
4. In the **Recognition type** field, select Bar code.
5. Select the Bar Code tab.

Note: **For InScript2 Users** – If you selected InScript2 in the **Recognition type** field on the Recognition tab, you cannot set the field values on the Bar Code tab. The InScript2 engine contains its own bar code support, and the properties for this must be set on the advanced properties dialogs for InScript2. For more information, see the Setting SEA (Bar Code) Sub Engine Properties section.

---

6. In the **Type** field, select the bar code type.
7. In the Orientation area, select the option that indicates the direction in which the bar code is positioned on the image. Options are **Horizontal** or **Vertical**.
8. Select the **Find On Page** check box to search for a bar code outside the recognition zone of the current field. The recognition engine will search the entire image until it finds a bar code of the specified type.

Note: **For InScript2 Users** – If you define a single field with an image clip that is the dimensions of the entire page, InScript2 will search the entire image and find the given bar code. In this case, you do not need to select the **Find on Page** check box.

Testing Bar Code Recognition

The bar code recognition engine is called separately from the other recognition engines in FormWare. Select the **Test** button on the Form Properties, Bar Code tab to execute bar code recognition for any fields on the current data form that are specified for bar code. A dialog displays the progress of the bar code recognition engine. When processing is complete, the results display in the bar code fields.

Note: **For InScript2 Users** – Because the Form Properties, Bar Code tab is not available for InScript2, test bar code recognition from the Field Properties, Test tab for the bar code field you created.

Testing bar code properties from the Form Properties dialog executes the bar code recognition process only for the current data form against the setup image. This is referred to as discrete testing. You can also perform comprehensive testing of multiple processes for the current data form on a batch of images from the Form Properties, Test tab. The comprehensive test allows you to see how the bar code properties are affected by another process, such as Image Enhancement.

Creating Mark Sense Recognition Fields

Mark sense recognition recognizes check marks, Xs, or other marks placed in check boxes. Mark sense recognition is also called Optical Mark Recognition (OMR). There are two types of mark sense recognition: single box and multi-box.

The ideal mark sense zone on a data form has plenty of white space around the boxes. However, this is not always possible. Try the following suggestions to improve recognition of a mark sense zone:

- Draw check boxes as tightly as possible on non-dropout forms where there are other form background features nearby. Draw check boxes a little larger on a dropout form than you would on a non-dropout form.

- Use Image Enhancement features only if the form text is as large as the check box, if form text is close to the check box, or if lines are close to the check box. Do not remove portions of the check box itself.

- Increase the Point Size Filter setting on the Removal Features Properties dialog if form text is interfering with mark sense recognition.

- If necessary, you can remove two successive sides of a check box without interfering with the recognition results.

- On noisy forms, where form text interferes with mark sense recognition, increase the Pixel Filter Size setting on the Removal Features Properties dialog.

- When possible, set the maximum number of check boxes allowed by setting the field length on the Recognition page to the maximum number allowed.

Understanding Mark Sense Return Data

You control the data returned when a mark is detected in a check box. In the case of single mark sense, you can return a Y if the box is checked and an N if it is not, or any character of your choice. For multiple mark sense, a series is set up to determine which of the check boxes are marked. This series can be either alphabetic or numeric. Therefore, you might receive 256 signifying that the second, fifth, and sixth check boxes had marks detected in them.

For single mark sense, the value you specify in the If Sensed field is returned when a mark in the zoned check box is detected. The value you select in the If Not Sensed field is returned when a

mark is not detected. You do not have to specify a value for both If Sensed and If Not Sensed fields. The If Not Sensed field does not apply to multiple mark sense recognition zones. The value you set in the Confidence field on the Field Properties, Recognition tab represents the percentage of the check box filled for a mark to be detected.

For multiple mark sense, the value you specify in the If Sensed field is the start character for the return values. The start character is assigned to the first check box configured in the zone. This value is then incremented and each incremented value is assigned to each successive check box configured in the zone. The keying order of the check boxes in the multiple mark sense zone is defined within the Multi-Mark Sense dialog. For example, if you configure five check boxes in the multiple mark sense zone and set an If Sensed value of M. Then, when marks are detected in the second and fourth boxes, an N and a P are returned.

Note: The number of boxes that can be included in a multi-mark sense field is 31. The "character if sensed" is used as an ASCII integer value, and simply incremented for each box. For example, if you use A as the character if sensed, then the 27th box (i.e., one after "Z") is the integer value 91, which is ASCII character "[".

The Data Length option setting on the Recognition page of the Field Properties dialog determines the number of return values stored for the Multi Box type for the mark sense field. If the data field is three characters long, then up to three return values are stored, even if the recognition engine returns four or more values.

Defining a Single Mark Sense Recognition Field

When using single mark sense, you zone a single check box and the recognition engine determines whether that check box is marked. Use single mark sense recognition if your form image contains a Yes and a No check box or if it contains a single check box as an answer to a question.

Think of the return data for a single mark sense recognition zone as a Boolean that returns only one of two possibilities, such as returning either a Y or an N. You can specify any two alphanumeric characters as the return for a mark sensed or not sensed.

To create a mark sense zone for a single check box on the setup image:

1. Open a job in Designer and select the data form on which you want to create a single mark sense field.

2. Create a new field and display the Field Properties, Recognition tab for that field.

3. Draw a recognition zone around the single check box on the setup image.

4. In the **Recognition type** field, select **Mark Sense** to enable the Mark Sense tab options.

5. In the **Data length** field, select 1, since the field will only contain one character of data.

6. In the **Confidence** field, select the percentage that the box needs to be filled for a mark to be detected in it. A value between five and ten percent is recommended.

7. Select the Mark Sense tab.

8. In the **Type** area, select the **Single Box** check box.

9. In the **If Sensed** field, type **Y** and in the **If Not Sensed** field, type **N**, or any other alphanumeric characters.

10. In the **Cross-out** field in the **Threshold levels** area, select a multiplicative factor that applies to multi box fields only. When multiple marks are detected within a Mark Sense field that are darker than the Black level, and they are greater than the Cross-Out factor multiplied by the

lightest mark, the box is considered crossed out and reported as unchecked. The Cross-out factor is only implemented for the InScript2 OMR engine.

11. In the **White** and **Black** fields, select the threshold levels that determine whether the box is checked. The white level is the level below which a box is considered unchecked, while the black level is the level above which a box is considered checked. If a level returned by the recognition engine falls between these two levels, the box is considered rejected.

12. Select the **Dropout boxes** check box if the production images are scanned with dropout ink. This means that the instructional and guiding marks on the form do not appear in the final, scanned image. If you select this option, the recognition engine does not search for the check boxes to guide the recognition, but rather for marks within the check box zones. When using dropout forms, scan an image that does not drop out the ink to use for design purposes as the setup image.

13. Select the **Save All** button on the Designer window toolbar to save your changes.

Defining Multiple Mark Sense Recognition Fields

Multiple mark sense recognition enables you to specify up to 31 different check boxes in one mark sense zone. These different check boxes are typically related. The recognition results depend on the field length and value specified in the **If Sensed** field.

Follow these steps to create a mark sense zone for multiple check boxes in one zone.

1. Open the job in Designer and select the data form where you want to create a multiple mark sense field.

2. Create a new field.

3. Draw a single recognition zone around all the check boxes on the setup image that you want to include in the multiple mark sense zone.

4. Display the Field Properties, Recognition tab for the field.

5. In the **Recognition type** field, select **Mark Sense**. This enables the Field Properties, Mark Sense tab for the field.

6. In the **Data Length** field, select the number of return values you want to store.

7. In the **Confidence** field, select the percentage that the box is filled for a mark to be detected in it. Values between five and ten are recommended.

8. Display the Field Properties, Mark Sense tab.

9. In the **Type** area, select the **Multi box** check box. This enables the **Min** and **Max** fields in the **Type** area.

   a. In the **Min** field, type the minimum value in a range of mark sense boxes checked. The values in the **Min** and **Max** fields define the acceptable range of boxes actually checked. When the number of boxes actually checked falls *below* this value the multi-mark sense field is rejected.

   b. In the **Max** field, type the maximum value in a range of mark sense boxes checked. The values in the **Min** and **Max** fields define the acceptable range of boxes actually checked. When the number of boxes actually checked rises *above* this value the multi-mark sense field is rejected.

10. In the **If Sensed** field, type 1, A, or another character depending on the return data you want to store for the start character.

11. In the **Cross-out** field in the **Threshold levels** area, select a multiplicative factor that applies to multi box fields only. When multiple marks are detected within a Mark Sense field that are darker than the Black level, and they are greater than the Cross-Out factor multiplied by the lightest mark, the box is considered crossed out and reported as unchecked. The Cross-out factor is only implemented for the InScript OMR engine.

12. In the **White** and **Black** fields, select the threshold levels that determine whether the box is checked. The white level is the level below which a box is considered unchecked, while the black level is the level above which a box is considered checked. If a level returned by the recognition engine falls between these two levels, the box is considered rejected.

13. Select the **Dropout boxes** check box if the production images are scanned with dropout ink. This means that the instructional and guiding marks on the form do not appear in the final, scanned image. If you select this option, the recognition engine does not search for the check boxes to guide the recognition, but rather for marks within the check box zones. When using dropout forms, scan an image that does not drop out the ink to use for design purposes as the setup image.

14. Select the **Multi Setup** button. The Multi-Mark Sense dialog displays the mark sense zoned area.

Note: The AutoFind feature is automatically applied to the check boxes displaying on the image. If boxes exist when you open the dialog for the first time, there is a set of defined check boxes for the zoned area.

15. Select the **Zoom In** button to increase the size of the image display or the **Zoom Out** button to decrease it.

16. Select the **Show Search Area** button to displays the search area. The search area is the number of pixels in X and Y coordinates beyond the check box. The X coordinate relates to the right and left sides and the Y coordinate relates to the top and bottom sides. For example, if the X and Y coordinates for a search area are 20 and 15 respectively, the left and right of the search area are 20 pixels larger than the mark sense box. The top and bottom sides of the search area are 15 pixels larger than the mark sense box. If a search area is not defined, this toolbar button is disabled. You can define a search area that is examined for non-dropout boxes. If a search area is defined, a box displays in blue.

17. Select the **Find** button to search the multiple mark sense zone for patterns representing check boxes and define those zones. This is also known as the AutoFind feature. AutoFind only finds check boxes that appear as non-dropout boxes in the image, either because it is a non-dropout form, or because you have drawn boxes in your setup form and scanned it as non-dropout. If you have a dropout image, you must manually draw in the boxes.

18. To resize a zone, select the **Select** button and use your mouse to change the zone size.

19. Select the **Define Check Box** button to draw a new check box zone.

20. Select the **Reorder Horizontal** button to change the check box zone keying (or searching) order horizontally. This process orders the check box zones from top to bottom, and if more than one zone is located on the same horizontal line, it orders them from left to right.

21. Select the **Reorder Vertical** button to reorder the check box zone keying (or searching) order vertically. This process orders the check box zones from left to right, and if more than one zone is located on the same vertical line, it orders them from top to bottom.

22. Select the **Set Keying Order** button to modify the keying order of the defined check box zones. The keying order is used to determine the value returned when a mark is detected in a check box. To reset the keying order, click the green numbers with the cursor in the order you

want. The cursor displays the next keying order number to be assigned. To reset the cursor to one, double-click outside the boxes.

23. Select the **Delete** button to delete the highlighted check box zone or **Delete All** to delete all of the check box zones.

24. Select the **OK** button when you are finished configuring and ordering the check box zones.

25. Select the **Save** toolbar button to save your changes.

Defining Removal Properties for Mark Sense Fields

You can set removal properties to remove check box lines and increase the accuracy of recognition. To use removal properties, you must specify on the form level whether the mark sense boxes are printed with dropout ink when the document is scanned. To do this:

1. Open a job within the Designer module.

2. Display the Form Properties, IPP tab for a form within the job.

3. Select the **Drop out ink** check box if the boxes on the scanned image were created using dropout ink. This is the only removal property you need to set on this page.

Defining Mark Sense Verification

If you process forms or surveys that contain a large number of mark sense boxes, you might want to use the mouse to verify mark sense recognition results. Mark sense verification allows a Completion operator to click with the mouse in a mark sense zone on the image to verify results, instead of entering or deleting a value in the corresponding data field.

The following are requirements of a mark sense verification application:

- The same Image Viewport contains a single group of mark sense boxes.

- The mark sense boxes within a group are mutually exclusive.

- Configure the mark sense fields as single mark sense (**Single Box** type) on the Mark Sense tab of the Field Properties dialog.

- Set the **If Sensed** field to X and set the **If Not Sensed** field to a "_" (space).

Mark sense verification occurs during the Verify mode of Completion. With mark sense verification, JobFlow presents all non-mark sense fields to the Completion operator in the New mode of Completion. When JobFlow performs the verify mode on the BDF, each group of mark sense fields is presented to the operator. The recognition engine assumes a group to be any single mark sense fields assigned to the same Image Viewport.

When using a job configured for mark sense verification, either a gray or red box displays around the mark sense boxes on the image. A red box indicates that a mark is detected in the mark sense zone, and the recognition engine assigns to the field the If Sensed value. A gray box indicates that a mark is not detected in the zone, and the recognition engine assigns to the field the If Not Sensed value. If a discrepancy is found, the operator uses the mouse to click in the box to change the data inserted in the field.

For example, if an image displays with a mark in box one, but a gray box displays around the zone indicating that the recognition engine found no marks in the zone. The operator can click on the zone to change the value to If Sensed and the zone changes to a red box frame. The field data is changed as well. When the group of check boxes is verified, the operator clicks on the Next Reject button, the last button on the Completion tool bar.

Use the following steps to configure a job for mark sense verification.

1. Open a job in Designer and display the form where you want to perform mark sense verification.

2. Set up your mark sense fields on the data form.

Note: When setting recognition zones for the mark sense fields, verify that each pair of mark sense fields you want grouped together has a unique Image Viewport.

3. Display the Form Properties, Image tab and specify the **Framed Zone** value in the View area.

4. Select one of the mark sense fields on the data form and display the Field Properties, Recognition tab for that field. Verify that the **Mark Sense** option is selected in the **Recognition type** field.

5. Display the Field Properties, Mark Sense tab. Verify that **Single Box** check box is selected as the **Type**. Type an X for the **If Sensed** field and a character space for the **If Not Sensed** field.

6. Display the Field Properties, Check Box Edits tab. Set the **Verify** field to Key for the mark sense field. Set the **Verify** field of all non-mark sense fields to **Skip**.

7. Repeat the previous steps for each of the mark sense fields in the verification job.

8. After the job is configured, create a JobFlow that includes the Recognition, Completion – New, and Completion – Verify tasks.

9. Select the **Save All** toolbar button to save your changes.

Testing Mark Sense Recognition

To test mark sense recognition for any single or multi mark sense field:

1. Launch the Designer module and display the Field Properties, Mark Sense tab for a mark sense field.

2. Select the **Test** button to load the mark sense recognition engine and test the mark sense properties.

3. If you do not receive valid results, check your mark sense options and try again.

Note: Mark sense recognition is particularly sensitive to registration. If you experience mark sense recognition errors, make sure that the images are registering properly.

Testing the Mark Sense properties from the Form Properties dialog, Mark Sense property tab executes only the mark sense recognition process for the current data form against the setup image. This is referred to as discrete testing. You can also perform comprehensive testing of multiple processes for the current data form on a batch of images from the Form Properties dialog, Test tab. A comprehensive test of the mark sense properties allows you to see how mark sense is affected by other processes, such as Image Enhancement.

Defining Recognition for Paragraph Zones

A paragraph field consists of one or more single-lined fields associated with a single, multi-lined image zone. Paragraph fields group individual fields together to form a logical, multi-lined field that preserves vertical white space. The paragraph field is most commonly used for address blocks.

To create a paragraph field:

1. Open a job the Designer module and define a field on the data form.

2. Move the image in the Image Viewport to display the paragraph you want to zone.

3. Select the paragraph field, and then zone the paragraph area on the image.

Note: When you define the paragraph zone on the image, make certain that no other zones lie within the paragraph zone area. Use the **Zone View** option to view all zones.

4. Double-click the field to display the Field Properties dialog.

5. Select the Recognition tab.

6. In the **Read Format** field, select the **Paragraph** option.

7. Copy the selected field so that the number of fields matches the number of lines within the paragraph on the image. For example, for a paragraph field with three lines, you would copy the field twice because you would need three fields for the three lines.

8. Display the Job Properties dialog, Image tab.

9. Select the **Adjust Field Zones** check box to cause character highlighting to be displayed during a Completion session.

10. Save your changes.

11. To test your paragraph field:

    a. Display the Form Properties dialog, Recognition tab for the paragraph field.

    b. Select the **Test** button. The recognition engine launches and the recognition results display in the fields.

    c. Save your changes.

    d. If your results are not correct, fine-tune the recognition options and retest the field.

12. To adjust the paragraph field options:

    a. Display the Field Properties dialog, Image tab for the paragraph field.

    b. Use the **Char W**, **Char H**, **Char Gap**, and **Line Gap** fields for paragraph field adjustment. The software attempts to auto-detect the correct values when you set these values to 0 (zero).

    c. If you adjust any field properties, do so on the header or first field. Then delete all other fields and copy the header field so that you have one field for each row. All fields used with the paragraph field MUST be identical.

    d. Save your changes.

Defining Recognition for Table Zones

A table field uses a super zone that includes more than one recognition zone on the image. It enables the recognition engine to perform column processing on form data. Table fields enable you to quickly create multiple fields and zones for data appearing in a table format on the image. The intersection of a row and column within a table field is called a cell.

For example:

	Column 1 ↓	Column 2 ↓	Column 3 ↓	Column 4 ↓
Row A →	Cell 1A	Cell 2A	Cell 3A	Cell 4A
Row B →	Cell 1B	Cell 2B	Cell 3B	Cell 4B
Row C →	Cell 1C	Cell 2C	Cell 3C	Cell 4C
Row D →	Cell 1D	Cell 2D	Cell 3D	Cell 4D

When you define a number of fields and add them to a table field, the recognition engine uses the vertical and horizontal position of the recognition zone to determine the correct location for data within the table.

Table fields are available with all FormWare recognition engines except Parascript. You must use the same recognition engine for all fields within a single table zone and the table field.

Note: You must create a recognition zone associated with a table field, or you will receive an error.

To create a table field:

1. Open an existing job inside Designer. The data form and setup image for this job display on the Designer window.

2. Move the image in the Image Viewport to display the area containing data you want in the table field.

3. Define fields on the data form for the first row in the table. These fields then act as column headings for the table. If you use the Nestor, AEG, or Mitek recognition engines, you must associate these fields with the area in the setup image that corresponds to the top of each column in the table field.

4. Continue associating each column heading in the setup image with a column-heading field on the data form, until complete. If you are using the InScript recognition engine, define the image zone for the heading field to encompass the entire column. This means that the recognition zone for the heading covers not just the column heading, but also all data in the column.

5. Duplicate the heading fields for each row of your table. To do this:

    a. Hold down the [Shift] key and use the left mouse button to select all cells in the heading row.

    b. Right-click to display the field context menu and select **Copy**.

    c. Click anywhere on the data form, right-click the mouse, and select **Paste** to paste the copied row of cells on the data form.

6. Move the cells to just beneath the column-heading row of cells on the data form.

7. Continue copying and pasting rows until all rows for the table field are defined. For example, if you have six rows on your document, duplicate the heading row five times.

8. Select the **Table Zone** button from the Field Type toolbar. The cursor changes to a cross hair (+).

9. Move the cross hair over the data form and click your mouse once.

10. A red, one-character field is created on the data form. This field is a placeholder for the table properties and is not assigned a field number. Most field properties are not available with the table field.

11. With the table field selected, create a super zone that encompasses all data in the table on the setup image.

12. Be certain that your table zone surrounds all of the table data.

13. When using the InScript recognition engine, copy a heading zone to the table field so the zones for all fields are the same.

14. Display the Job Properties dialog and select the Image tab.

15. Select the **Adjust Field Zones** check box to display character highlighting during Completion.

16. Select **Save** to save your changes.

17. To test the table field, display the Form Properties dialog and select the Recognition tab.

18. Select the **Test** button. The specified recognition engine loads and places the recognition data in the fields.

Note: Do not use the Field Properties **Test** button to test a table field.

19. If your results are not correct, fine-tune the recognition options and retest the field.

Note: The InScript recognition engine does not support multiple column processing. It supports tables through use of the "TABLE" rule. To use this rule, zone each column as a separate table zone and associate the columns into a table with the rule.

20. To adjust the table field options, display the Field Properties dialog, Image page for the table field. Use the **Char W**, **Char H**, **Char Gap**, and **Line Gap** fields for table field adjustment. If you set these options to zero, the software attempts to auto-detect the correct values.

Note: If you adjust any field properties, do so on the header or first field. Then delete all other fields and copy the header field so that you have one field for each row. It is extremely important that all fields used with the table zone be exactly the same.

21. Select **Save** on the Designer window to save your changes.

Defining Recognition Using Column Grouping

Column Grouping allows you to design a columnar relationship between multiple, previously unrelated fields on a data form. This is an easier to configure and more precise analog to the Table Zone method of enabling table processing.

Caution: If you have an existing FDF that contains column groups defined using the OLD column method (i.e., using the "red" table zone definitions), you CANNOT add new column groups to this FDF using this new method. Mixing new and old style columns within a single FDF results in an error message when you attempt to run the recognition engine on that form.

Follow these steps to define a column grouping on a data form.

1. Open an existing job in Designer.

2. Open an existing form or create a new one that contains at least two fields.

3. On the data form, select the fields to include in the column grouping with the mouse using [Shift-Click].

4. Right-click the mouse within the data form area to display the context menu.

5. Select the **Select Column Grouping** option to display the Column Grouping dialog.

6. Select the **New Group** button to display the Enter Name dialog.

7. In the **Name** field, type the name of the new column group and then select **OK** to return to the Column Grouping dialog. The name you typed displays in the **Existing Groups** field.

8. Select the **Add to Group** button to add all fields in the **Selected Fields** list as members of this group. The selected fields display in the **Members** list.

9. To delete the group in the **Existing Groups** field, select the **Delete Group** button.

10. To give the group a new name, select the **Rename** Group button. Type the new name in the Enter Name dialog and select **OK**.

11. To remove a field from the group, highlight the field in the **Members** list and select the **Remove From Group** button.

12. Select **OK** to accept your changes and return to Designer.

13. Save your changes to the job.

Defining Full Text Recognition

Full Text Recognition uses an OCR (Optical Character Recognition) engine to convert scanned documents into word processing, spreadsheet, or database files. This eliminates the time-consuming, physical retrieval of stored files.

For example, you can use full text recognition to archive legal documents by customer. Scan the documents and create a batch of related images for each customer. Use full text recognition to convert the batches into PDF files and output one PDF file per batch. You now have PDF files of documents by customer that you can access using Adobe Acrobat. You can also index these PDF files, making information search and retrieval a much simpler process. To index the PDF files, start each batch with a header page containing information such as Customer Name, ID, Attorney, etc. and identify this information as index fields.

Note: To use full text recognition, input images must be 200 DPI or 300 DPI black and white images with machine print only. Recognition is best on text size where DPI x Point Size>2400 (i.e., at 200 DPI, the print size is larger than 12 point, and at 300 DPI, print is larger than 8 point).

Creating a Full Text Recognition Job

The following steps show you how to create a job in Designer to generate a recognition output file in PDF format for each batch of images processed.

1. Open Designer and create a new job by selecting the **New** button on the toolbar. The New Job Name dialog displays.

2. Type the name of the job in the **Job name** field and select **OK**. The Job Defaults dialog displays.

3. Select **OK** to use default settings on the Job Defaults dialog. This opens the job within Designer.

4. On the Designer window, verify that the **View Tests in Completion** button is *not* selected.

5. Select **Save All** on the Designer window toolbar to save your job.

6. If you are processing foreign language documents, create a job INI file in the \FormWare\Ini\Jobs folder. Add the LANGUAGE = <language name> parameter, where <language name> is the appropriate language name file you are using. See the Creating a Job INI File section for more information on creating an job INI file. See the Using Full Text Recognition with Foreign Languages section for information on the available language files.

7. From Designer, display the Job Properties, Scan tab.

8. On the Scan tab, select the **Properties** button to access Scan Designer if you need to create or import images for this job using a scanner. See the Defining Scan Settings section for more information.

9. On the Job Properties, IPP tab, select the **Properties** button to display the Image Pre-Processing Options dialog and *deselect* all options on this dialog.

10. On the Job Properties, Recognition tab, select the **Full Text** button to display the Full Text Properties, Export tab for the job.

11. In the **Mode** field, select file export type. Options are:
    o   **One File Per Image** – **Export Path** options are *enabled*. One output file is created for each input image. The output path and file extension are determined by the settings within the **Export Path** area.
    o   **One File Per Document** – **Export Path** options are *enabled*. One output file is created for each input document. The output path and file extension are determined by the setting within the **Export Path** area. Functionality of this option is identical to the **One File Per Image** option with the exception that only one output file is generated *per document*. The base name the output file is the same as the *base name of the image that defines first page of the document*.
    o   **One File Per Batch** – **Export Path** options are *disabled*. One output file is created for each input batch. The output file is placed in the exact defined in the **Path** field. It will bear the name of the root batch file processed with the file extension defined in the **File Ext** field. Options for appending paths and file extensions is not available for batch files as it could cause potential problems within the FormWare Database or within InputAccel.

12. In the **Format** field, select the appropriate export file format (e.g., Adobe PDF "Normal"). The **File Ext** field displays the file extension associated with the export format. See the Understanding Full Text Recognition Output Files section for a list of available output formats and their extensions.

13. In the **Export Path** area, select the **Browse** button adjacent to the **Path** field to define the location to which Full Text recognition output files will be exported.

14. Select one of the following options to define the export path:
    o   **Use Exact Export Path** – Sends export files to the exact path defined in the **Path** field.
    o   **Use Image Path** – Sends export files to the path defined in the **Path** field *plus* the current source image or document path ("EXPORT PATH" + "IMAGE PATH"). For example, if the source image path is *c:\invoices* and the export path is *c:\formware\results\fwftclnt t*hen the output files are written to *c:\formware\results\fwftclnt\invoices*.

Note: Because the default is to Use Exact Export Path option, the output files are placed into the defined export path and the existing file extension is replaced with the extension specified in the File Ext field. If you have files that bear the same root filename and either a different file extension or a different sub-directory, you can potentially overwrite files using this option. Instead, select the Use Image Path option to output files with the same root name to unique paths, thereby guaranteeing that the files will not be overwritten.

15. Select one of the following options to define the file extension used for all export files:

   o **Overwrite File Extension** – All export files will use the exact file extension as defined in the **File Ext** field. This enables you to define the export filename the same as root name with the defined file extension (e.g., *ImageName.tif* would be exported to *ImageName.txt*).

   o **Append File Extension** – All export files will append the file extension defined in the **File Ext** field to the original source image files extension. For example, if the source image or document name is *Invoice.tif*, and the **File Ext** is *txt*, then the output file will be named *Invoice.tif.txt*.

Note: Selecting the **Append File Extension** option enables you to preserve the output extension defined by the export type without overwriting export files with incremental file extensions that reside in the same directory (e.g., InvoiceName.001, InvoiceName.002).

   o **Leave File Extension** – All export files will use the original source image file or document file extension only. For example, input source file name is *Invoice.tif*, the output file will be named *Invoice.tif*.

16. Select the **Styles** check box to enable the **Style Prefix** field. This feature preserves the original document format by using styles when you export in a supported format, such as a word-processing format.

17. In the **Style Prefix** field, type the characters to enter at the beginning of the style name. The recognition process automatically assigns style names.

18. Select the **Output Pictures** check box if your images contain graphics and the selected export file format supports them. In most formats the graphics output at the end of the document.

19. Select the **Page Mirror** check box to preserve the inside and outside margin measurements for facing pages.

20. Select the **Unidentified Forms** tab. Set these values if you are creating a job to process images as both data and full text capture. Full text capture only jobs do not use these settings. On this tab, you set the processing options for all images that fail Form ID as "full text capture" documents.

Note: The **Default to job properties** check box on the Form Properties dialog for full text recognition uses the settings on this tab.

21. Select the **Process Unidentified Forms** check box to enable the remaining fields on this tab. This indicates that you intend to process any forms that fail Form ID as full text capture documents. You assign Form ID to your data capture form.

22. In the **Job Image Quality** area, select one or more of the following options for this full text job:

   o **Poor Image** - Slows the OCR processing to improve the processing of poor quality images for this full text job.

- o **Dot Matrix** - Enables the recognition engine to process dot matrix print for this full text job.
- o **News Print** - Enables the recognition engine to process newspaper text for this full text job.
- o **Fax** - Enables a time stamp read engine for this full text job.

23. Select **OK** to accept your changes and return to the Job Properties, Recognition tab.

24. Select the form name in the Designer window Job Browser to display the Form Properties dialog.

25. Select the Image tab and specify a setup image.

26. Select **IPP** button to test IPP against the setup image. You are testing with the form level default settings.

27. Review the setup image in the Image Viewer and select **OK** to accept the enhanced image.

28. On the data form, define one text field using the **Any Type** button on the Field Types toolbar. This enables you to run a test on your setup image from the Recognition page.

29. From the Form Properties dialog, select the Recognition tab.

30. Select the **Full Text** button to display the Full Text Properties dialog for the form.

31. Select the **Full Text Recognition** check box to enable all fields on this tab. This defines the form as a Full Text Capture document.

32. In the **Form** area, select from the following options.
- o **Poor Image -** Slows OCR processing to improve results from poor quality images.
- o **Dot Matrix** - Enables the recognition engine to process dot matrix print.
- o **News Print** - Enables the recognition engine to process newspaper text.
- o **Fax** - Enables a time stamp read engine.

33. Select **OK** to accept these settings and return to the Form Properties dialog.

34. Select **Save All** on the Designer window toolbar to save your changes.

Understanding Full Text Recognition Output Files

There are two ways to associate full text documents with index data:

- If the page with index data is also to be captured as full text, you can capture the zoned form index fields as OCR or KFI data and export them using FormWare. You can specify the same form type as a "full text form" for full text capture.

- If the index data is on a form preceding the full text pages (which may be treated as attachments), this should be processed using document mode. The "Start of Document" form can be the index data and the remaining pages can be output as full text. Full text data output in document mode has the same name as the first image in the document, so you can associate the full text output with the start of document form data.

Full text recognition supports document conversion to many of the most popular formats. The conversion process also preserves formatting with document styles if the output format supports styles. Following is a list of the available format options:

Application	Extension	Application	Extension
8-Bit Text	.txt	MS Word 6.0 & 7.0	.doc

Application	Extension	Application	Extension
Adobe PDF "Normal" with images for suspect words	.pdf	MS Word 97	.doc
Adobe PDF "Normal"	.pdf	MS Works	.rtf
Adobe PDF original image on top of invisible text	.pdf	MultiMate Advantage	.doc
AMI Pro version 2.0 & 3.0	.sam	Postscript	.ps
ASCII "Smart" DOS	.txt	Prof. Write 2.0 & 2.2	.doc
ASCII CSV with Tabs	.txt	Quattro Pro	.wb2
ASCII CSV	.txt	RTF Mac	.rtf
ASCII Stripped DOS	.txt	Simple (galley) HTML	.htm
dBASE IV	.dbf	"Smart" ASCII	.txt
DCA RTF	.rtf	Standard ASCII	.txt
Display Write v5	.rtf	Standard DOS ASCII	.txt
Excel 97	.xls	Stripped ASCII	.txt
Excel Mac v3.0-7.0	.xls	Unicode Text	.txt
Excel v3.0-5.0	.xls	Windows Write	.doc
FrameMaker	.mif	Word for Windows 2.x	.doc
Interleaf	.wps	WordPerfect 4.2 & 5.1	.wpf.
ISO Text	.txt	WordPerfect 6.0-8.0	.wpd
Lotus 1-2-3	.wk1	WordStar	.wsd
Lotus Word Pro	.lwp	xdoc Lite	.doc

Using Full Text Recognition with Foreign Languages

In our global economy, organizations frequently manage documents written in languages other than English. Full text recognition can process those documents by setting a particular job level language parameter. Specify a language to the engine by adding the LANGUAGE= parameter to the job INI file. This instructs the job to recognize and support only a single language. If you do not specify a language, US – English is used as the default.

The following is a list of supported language files within full text recognition:

Language	Filename	Language	Filename
ASCII English	asciieng.lng	Hungarian	hungar.lng
Czech	czech.lng	Italian	italian.lng
Danish	danish.lng	Norse	norsk.lng
Dutch	dutch.lng	Polish	polish.lng
English	english.lng	Portuguese	port.lng
Finnish	finnish.lng	Russian	russian.lng
French	french.lng	Spanish	spanish.lng
German	german.lng	Swedish	swedish.lng
Greek	greek.lng	Turkish	turkish.lng

Testing Full Text Recognition

Test your property settings against the setup image and a batch of images. When testing multiple image files, use a large test sample that adequately represents the variations in the documents you are capturing.

Testing Full Text Recognition on a Setup Image

Follow these steps to test full text recognition on a setup image:

1. Display the Form Properties, Recognition tab and select the **Test** button. The recognition progress window displays the status of InScript (default engine) and FWFTClnt engine processing. The test produces a converted file and exports it to the defined destination. Note the processing time for analysis purposes.

2. Open the file from the associated application and review the conversion results.

3. Modify property settings and retest if necessary.

Testing Full Text Recognition for a Batch

Follow these steps to test full text recognition on a batch:

1. Display the Job Properties, Test tab.

2. Select the **Recognition** check box.

3. Select the **Browse** button and find a BDF to test.

Note: BDFs must contain a job name for Cleanup to run successfully at the end of the test. Select the **Edit** button to view the contents of the BDF.

4. Select the **Test** button. The recognition progress window displays the status of InScript (default engine) and FWFTClnt engine processing. The test produces a converted file and exports it to the defined destination.

5. Open the file from the associated application and review the conversion results.

6. Modify property settings and retest if necessary.

Defining Free Form Recognition

Free Form Recognition is technology for locating and extracting data from both structured and unstructured documents. Use Free Form when the form data does not reside in a static location from one image to the next. Free form fields are fields that you create to identify shifting data and enable search processing to occur on the form. When using Free Form recognition, you create the form definition file (FDF) in Designer for the form classes you process (e.g., a single page invoice), detailing the fields to be captured and properties for each field. Then, at the field level you define Free Form fields that include the pattern-based information (keyword context and data pattern) for determining how to locate and extract the data.

Understanding Free Form Processing

Free Form processing is embedded in FormWare's Recognition clients. It makes use of full-page, recognized text and location information. Free Form includes support for a graphics-based

display for creating, training, and testing of keywords, data patterns, and relative data search areas (relative clips).

You define the Free Form keyword search list within Designer. Recognition sets the actual location of data for the fields. When Completion displays a form, it dynamically selects the character coordinates for each field on each image. The Free Form Properties dialog includes zoom, scroll, and drawing capabilities. This image display window enables you to graphically define (draw) the keyword box, data pattern box, and relative data search area box.

When you use Free Form, you create a job in Designer containing one or more form types. You can include any combination of fixed and Free Form field types within a single form, up to the limit imposed by the OCR file. When creating zones for Free Form fields, some or all of the zones may need to be sized to include the entire page for those invoice fields whose positions cannot be pre-determined. You assign each relevant data field a corresponding keyword on the Free Form Properties dialog.

When using Free Form, Recognition processes as follows.

1. Recognizes each field twice, once if the field type is other than Any Type and once for Any Type.

2. Returns raw data and keyword results.

3. Applies Free Form parameters first *excluded* keywords and parameters, then *included* keywords and parameters.

4. Applies rules.

5. Applies VBA.

Because the forms you process using Free Form are not uniform in appearance or data location, it can be difficult for FormWare to identify them correctly. To facilitate Form ID, add a FORCE_UNKNOWN=<form name> parameter to the System.ini to force identification for all unknown form types to a specific number. This allows a limited mixing of fixed forms and Free Forms during the scanning process.

Creating a Free Form Field

The options on the Free Form Properties dialog enable you to recognize Free Form data by specifying keywords and associated data coordinates, value ranges, and other information.

Perform the following steps to define a Free Form recognition field in Designer.

1. Open a job and form within Designer.

2. Display the Form Properties, Image tab and specify a setup image for the form.

3. Add a new field to the data form and draw an associated zone.

Note: You can select any field type. FormWare will recognize the field as the type you selected and as an Any Type field. Double recognition improves the recognition accuracy of Free Form fields. The Any Type data is used to find keywords and is the initial version of the data to be extracted from the field. Once the data is extracted, its coordinates are used with the assigned field type Recognition to retrieve the more correctly segmented and recognized data.

4. Display the Field Properties, Recognition tab for the field.

5. Select Machine print from the **Recognition type** drop down list.

6. Select the **Variable** option from the Read Length area. Although making this selection is optional, it will provide you with optimum results.

7. Select the **Free Form Properties** button. This runs the Free Form process that gathers OCR information for the form based on the selected setup image. When this initial process is complete, the Free Form Properties dialog displays showing the zone area in its image display area.

8. To view fields from images other than the setup image in the Free Form Properties dialog image display, select **Browse** to display the Open dialog.

9. Select a BDF file in the Open dialog and select **OK** to open that dialog in the Free Form Properties dialog. The name and path of the BDF displays in the Browse bdf's area above the **Browse** field and the name of the first image in the BDF displays in the **Browse** field.

10. Select the arrows in the toolbar to scroll to the previous and next images in the BDF. Results are updated in the image display.

11. Add keywords as needed using the method described in the Creating Keywords and Data Patterns section below.

12. Add columns as needed using the method described in the Setting Column Properties section below.

13. If you want to disable Free Form processing for the field, select the **Disable Free Form** check box. Although this option disables Free Form, it maintains any keyword grid entries for possible later use.

14. If you want to exclude keywords, select the **Advanced** button to display the Advances Settings dialog. Refer to the Excluding Keywords for more information.

Note: On the Advanced Settings dialog, General tab, the **Include common digit substitutions in data Regular Expressions** check box is selected by default for newly-designated Free Form fields. For backwards compatibility, this check box is unchecked by default for existing free from applications. Refer to the Managing Numeric Substitutions section for information on the functions of this checkbox.

---

15. If you want to remove all keyword grid entries, select the **Clear** button.

16. If you want to delete a row from the grid, right-click on the row and select **Delete** from the context menu.

17. Select **OK** on the Free Form Properties dialog to confirm your changes and return to Designer.

Creating Keywords and Data Patterns

Free Form includes the process of teaching FormWare to locate a keyword and search for data using that keyword. Perform the following steps to create and teach a new keyword.

1. Open the Free Form Properties dialog for any field and associated zone on which you plan to use Free Form. Be sure the associated zone contains at least one word you can use as a keyword.

2. Using the scroll bar and the **Zoom-in** and **Zoom-out** buttons (on the toolbar), locate the keyword and recognition area in the image viewport.

3. If the grid control is *not* currently empty (i.e., there are keywords that already exist in the grid), use the mouse and left-click anywhere within the next available empty row in the grid. This enables **Draw the Keyword to Search for** (key icon) button.

4. Select the **Draw the Keyword to Search for** button. Cross-hair lines display for drawing the keyword zone.

5. Using your mouse, draw a zone around the keyword you want to create. Draw the zone just tightly enough to *not* include any other characters or black blobs within the drawn zone area. Upon releasing the mouse, a red border displays tightly around the keyword.

6. Select the **Draw the Data to Search for** (123 icon) button to draw the data zone around the desired data pattern. This is usually different than the keyword area because Free Form uses the keyword to locate the data pattern. When you release the mouse, two borders display: blue is the data area and green is the relative clip (relative data search) area. The data pattern captured in the blue area will be used to create a regular expression for the Free Form data search.

7. The relative data search area is the actual area used to search for the data associated with the keyword. This is the area (in pixels) *relative to* the location of the upper left corner of the first character in the keyword found in the keyword search. The coordinates of this area appear in the **X**, **Y**, **Width**, and **Height** columns of the Free Form grid.

8. If you want to create a keyword for a field that contains no data on the training image, select the **Draw the relative clip to search for data** button after you have drawn the keyword search area. Use this option only if you have an empty area on your form that *does not currently contain data*, but that you know will contain data in production and should be used for a data search. The relative clip area is the only zone you can alter in the image display area. Using the drawing tool, you can resize or relocate the relative clip area. You cannot use the drawing tool on an existing keyword zone or data search zone.

9. In the grid portion of the Free Form Properties window, data fills in for the keyword you have just created. Free Form populates the grid based on the location of the keyword zone, the size of the data search area, and the type of data that you have included in the data search area.

Modifying Keyword Properties

Free Form populates all fields in the current row of the Free Form Properties dialog grid when you create a keyword in the image display area. Each populated row contains default values based on the keyword, data pattern, and relative data search zones you have defined. The list of keywords auto-sorts to display them in order of length, longest to shortest. This optimizes the search algorithm.

You can modify the grid values using the Free Form Keyword Properties dialog and reapply the settings to one or all of the keywords on the grid. However, if you change the coordinate settings (i.e., X, Y, Width, Height), you will alter the location of the relative data search area. If you really do want to change the location of the relative data search area, you will obtain the most accurate results by redrawing the zone in the image display area.

Perform the following steps to modify the properties of a keyword.

1. Open the Free Form Properties dialog for any field and associated zone on which you plan to use Free Form.

2. On the Free Form Properties dialog, right-click on the row in the grid that you want to change to display a context menu.

3. Select **Properties** from the context menu. The Free Form Keyword Properties dialog displays.

4. To change the keyword, type a new keyword in the **Keyword** field on which the recognition engine will search.

Caution: Use only alpha or numeric character when creating keywords. Many special characters have specific meaning in regular expression syntax and can cause the search algorithm to produce incorrect results or to fail.

---

5. In the **Keyword Confidence** field, select the percentage value that indicates the confidence threshold for the keyword search. This value indicates the percentage of characters out of the total number of possible characters that must successfully match the keyword. For example, if the keyword is TOTAL, the best match string result is TOTIL, and the **Keyword Confidence** setting is 80 percent, the match will be successful (four out of five characters matched correctly).

6. In the fields of the Relative Data Search Area, type the coordinates of the search area in which the recognition engine will look for the data pattern defined by the regular expression. This search area (in pixels) is relative *to* the location of the upper left corner of the first character in the keyword found in the keyword search. The **Width** and **Height** settings define the width and height, in pixels, of the search area. The **X** and **Y** pixel coordinates define the offset from the keyword of where the search area will be located. The **X** and **Y** values can be either negative or positive values.

7. As noted above, if you really do want to change the location of the relative data search area, you will obtain the most accurate results by redrawing the zone in the image display area.

8. Type the minimum and maximum number of characters allowed for a correct match by the search process in the **Min** and **Max** fields in the Characters in Data Options area. These values should match any indicated in the regular expression.

9. In the Data Pattern as Regular Expression field, type a regular expression pattern for the recognition engine to use when searching for the field value. You can expand the regular expression beyond the default entry created when you draw the keyword and data search areas. You should be familiar with regular expressions before altering the default entry.

Caution: To avoid the problems caused by special characters having a specific meaning in regular expression syntax, precede these characters at the beginning of a regular expression by inserting a \ (back slash) character. In the expression \$[0-9]{8,10}, the back slash indicates that the $ is actually part of the search pattern itself rather than having a special meaning.

---

10. Select the **Keyword** checkbox and/or the **Data** checkbox from the Accept First Match group to return only the first instance only of the keyword or data found. By default, these options are not selected. If you do not select these options, Free Form evaluates all instances of valid keyword matches.

11. Select the **Case Sensitive** check box to indicate that the data search should be case sensitive. When drawing a keyword zone and the associated data search areas, by default this option is not selected. You must modify the setting to enable case sensitivity.

12. When you do not select the **Case Sensitive** check box, keyword searching is fully case insensitive for *each* character in the keyword. If you do select the **Case Sensitive** check box, Free Form may still successfully match to a *mixed* upper and lower case word, even when you create the keyword as a single case word. This can occur for any character that *looks like* its lower case version, or even a lower case version of a different character. The following pairs of characters are examples of this situation.

- C, c
- L, l
- I, i
- G, c

13. To avoid this, you can *increase* the confidence value (closer to 100%) if you really do not want such a match to occur.

14. If you want to delete the currently selected row in the grid, select the **Delete** button.

15. If you want to apply your changes only to the keyword you are modifying, select the **Apply** button. If you want to apply your changes to all records in the grid, select the **Apply All** button. Selecting either the **Apply** or **Apply All** buttons confirms your changes and returns you to the Free Form Properties grid.

After you create or test keywords on the Free Form Properties grid, the following columns display recognition test results.

- **Key Found** - Displays the keyword string result of the recognition test.

- **Data Found** - Displays the data string associated with the keyword.

- **Conf** - Displays the confidence level of the test results. The confidence value is computed based on the number of characters out of the total number of characters possible that successfully match the keyword. For example, if the keyword is TOTAL, the best match string result is TOTIL, and the Confidence Threshold value is 80 percent, the match will be successful because four out of five characters matched correctly.

Searching with Regular Expressions

When you want to use only a regular expression as the search pattern for your data, you can create a line in the Free Form Properties grid that contains a regular expression, but does not contain a keyword. In this case, the entire image zone is searched for data that matches the regular expression criteria. You cannot use relative clip coordinates without a keyword, so those options are grayed out in the Free Form Keyword Properties dialog when the Keyword field is blank.

Perform the following steps to create a regular expression search pattern.

1. Open the Free Form Properties dialog for any field and associated zone on which you plan to use Free Form.

2. On the Free Form Properties dialog, right-click on a blank grid line in the grid control to display the Free Form Properties context menu.

3. Select the **Properties** option to display the Free Form Keyword Properties dialog.

4. Leave the **Keyword** field blank.

5. In the Characters in Data area, select values in the **Min** and **Max** fields to indicate the minimum and maximum number of characters in the data for which you are searching.

6. In the **Data Pattern as regular expression** field, type the regular expression to indicate the pattern of data for which you are searching.

7. Select the **Enable First Match** and **Case Sensitive** check boxes, if applicable.

8. Select **OK** to return to the Free Form Properties dialog. A record displays on the grid without a keyword in the **Keyword** field.

Managing Numeric Substitutions

When creating data regular expressions to search for digits in new Free Form fields, you do not need to specify all the substitution error characters. For new Free Form fields, select the Include common digit substitutions in data Regular Expressions checkbox on the Free Form Advanced Settings dialog, General tab. Free Form Recognition automatically adds common numeric substitution errors to any digit or digit range in the data regular expression, although the automatic substitutions are not displayed. The check box applies to the all data regular expressions for the entire field.

The following tables show the automatic substitutions Free Form uses.

Digit	Default Substitutions per Digit
0	Uppercase and lowercase o
1	Uppercase and lowercase i, lowercase l
2	Uppercase Z
4	Lowercase q
5	Uppercase and lowercase s
6	Lowercase b
8	Uppercase B
9	Lowercase g

Category	Expansions of Regular Expression Specs
single digit	Example: [A-Z12] expands to [A-Z12iIlZ]
digit range	Example: [A-Z2-4] expands to [A-Z2-4Zq]
character class "[:digit:]"	Expanded as if it were 0-9. Example: [A-Z[:digit:]] expands to [A-Z[:digit:]oOiIlZqsSbBg]

Creating Variations of Keywords

You can create variations of a keyword to better enable your application to capture the correct data. For example, if you want to capture an invoice number, you can define multiple keywords for variations on the word invoice and other indicators of an invoice number (e.g., invoice, inv., inv. no., #, number, no.). If no keywords are found during the actual recognition process, the field is rejected.

Perform the following steps to create variations of a keyword.

1. Open the Free Form Properties dialog for any field and associated zone on which you plan to use Free Form.

2. Using the Free Form Properties dialog, create a keyword for which you want to make variations.

3. Select the blank row immediately below the keyword grid row you created in step two and right-click to display the context menu.

4. Select **Properties** from the context menu. The Free Form Keyword Properties dialog displays.

5. Type the variation of the keyword you want to add into the **Keyword** field. The remaining fields auto-populate with the values of the previous keyword.

6. Select **Apply** to return to the Free Form Properties dialog and add your keyword variation to the grid.

Creating Multi-line Keywords

You can create a keyword search for multi-line keywords that could search for one word on one line, followed by a different word on the next line. Perform the following steps to create a multi-line keyword.

1. Open the Free Form Properties dialog for any field and associated zone on which you plan to use Free Form. Be sure the associated zone contains the words you plan to use as a multi-line keyword.

2. Using the scroll bar and the **Zoom-in** and **Zoom-out** buttons (on the toolbar), locate the keywords and recognition area in the image viewport.

3. If the grid control is *not* currently empty (i.e., there are keywords that already exist in the grid), use the mouse and left-click anywhere within the next available empty row in the grid. This enables **Draw the Keyword to Search for** button.

4. Select the **Draw the Keyword to Search for** button. Cross-hair lines display for drawing the keyword zone.

5. Using your mouse, draw a zone around the keywords you want to use as a multi-line keyword. Upon releasing the mouse, a red border displays tightly around the keywords.

6. Select the **Draw the Data to Search for** button or **Draw the relative clip to search for data** button, depending on the type of clip you want to create, and draw the area for your data clip. In the grid portion of the Free Form Properties window, data fills in for the keyword you have just created.

When you draw a multi-line keyword zone, Free Form inserts a *line feed* separator \n between the two keywords. You can view this separator in the Keyword and Key Found columns of the grid. You can insert any number of \n separators to search for any number of lines of keywords. However, multiple \n separators in a row and \n separators at the beginning or end are not acceptable.

You can also add a multi-line keyword using the Free Form Keyword Properties dialog. To do this, type an \n in the keyword where the new line should begin. For example, Unit\nAmount would act as a two line keyword.

Excluding Keywords

Use the Exclude Keywords tab to create a list of word to be excluded from Free Form processing. During Free Form, the program searches for the keywords to exclude *before* searching for the keywords in the Keyword Properties grid. Your application eliminates every occurrence of an excluded keyword from the Recognition data before any other Free Form processing.

Free Form eliminates excluded keywords only if they are found with a confidence level above the threshold you specify. For example, if your form is an invoice and you want to find an invoice number using the keywords shown on the Free Form Properties dialog but want to exclude keywords assigned to data you do not want included in the Recognition results, list the keywords you want to exclude on the Exclude Keywords tab.

Perform the following steps to exclude a keyword.

1. Open the Free Form Properties dialog for any field and associated zone on which you plan to use Free Form.

2. On the Free Form Properties dialog, select the **Advanced** button. The Advanced Settings dialog displays.

3. Select the Exclude Keywords tab.

4. Select the **New Exclusion** button to display the Exclude Keyword Properties dialog.

5. In the **Exclude Keyword** field of the Exclude Keyword Properties dialog, type the keyword you want to exclude.

6. In the **Confidence Threshold** field, select the percentage value that indicates the confidence threshold for the keyword search. This value indicates the percentage of characters that must successfully match the keyword out of the total number of possible characters.

7. Select the **Case Sensitive** check box if you want the excluded keyword search to be case sensitive.

8. Select **OK** to return to the Exclude Keywords tab.

Note: Optionally, type the height (in pixels) of characters to exclude in the **Exclude keywords with height** field. Free Form excludes any characters greater in height than this value from the Recognition results. Free Form includes any keywords that are smaller than the value. The default value for this setting is 50.

---

9. Repeat steps 4 through 8 to add any additional keywords to the exclusion list.

10. To delete a keyword from the list, double-click on the keyword and display the Exclude Keyword Properties dialog. Select the **Delete Row** button.

11. To clear the entire grid, select the **Clear Grid** button on the Exclude Keywords tab.

12. Select **OK** to confirm your changes and return to the Free Form Properties dialog.

Setting Column Properties

You can create Free Form recognition zones for data columns on a form using Free Form Recognition. This functionality accommodates large columns of data that appear on forms, such as an invoice, and locates the data in multiple rows within those columns.

Free Form column functionality is available for all FormWare supported Recognition clients. Free Form column functionality includes the following features.

- Multi-line column titles (keywords)
- Multi-page columns
- Ability to locate the end of a column

The relative clip (data search) area defines a search location for the first row (cell) of data in the column. This area must be long enough vertically to include the first non-empty row in which you expect to find data. If you anticipate that the entire column except for the last row could be blank, specify a relative clip area height that is roughly the height of the entire column. After Free Form finds the first data item in the first row, all other rows are automatically located by the Free Form algorithm.

Creating Column Fields

To set up a column header field and designate it as the main Free Form field within a table zone, you must create a table and set of column fields on the data form to correspond to a table or a single column on the image. Even if you want to process only a single column of data, you must set up a table field for the column.

Perform the following steps to create a column field and apply Free Form Recognition.

1. Open your job in Designer.

2. Create a field for the first row in the column that you will later configure as the column header field.

3. Draw the Recognition zone on the image to encompass the keyword at the top of the column and all data rows to be located on the form. If the column can be located anywhere within the table, draw the image zone to encompass the entire table. If the column is fixed, draw the zone to be slightly larger than the single column you are working with, including the keyword.

Caution: You must draw the image zone large enough to locate and process all rows within the column.

---

4. Cut and paste the column header field to create fields for all the other rows in the column. Ensure that the Recognition zone is identical for all rows. Do not change the image zone coordinates after the cut and paste operation.

5. When cutting and pasting fields on the data form, ensure that all the fields in the column are vertically adjacent (touching, but not overlapping). Additionally, make sure that each field is directly beneath the field above so there is no horizontal displacement or shift between the column fields. All of the fields must be perfectly aligned on the data form or Free Form column processing will not work correctly.

6. Double-click the Free Form column header field to display the Field Properties dialog for that field.

7. Select the Recognition tab and then the **Free Form Properties** button to display the Free Form Properties dialog.

8. Create a keyword and specify a relative data search area as described in the Creating Keywords and Data Patterns section above.

9. The keyword is typically the title of the column and can be a multi-line keyword that is common for column fields. Refer to the Creating Multi-Line Column Titles section below for more information about creating multi-line Column Titles.

10. The relative data search area defines a search location for the first row of data in the column. Ensure that this area is vertically long enough to encompass the first non-empty row where you expect to find data. If you anticipate that the entire column except for the last row could be blank, specify a relative data search area height that is roughly the height of the entire column.

Note: You do not need to configure separate keywords and relative data search area coordinates for rows 2 through 'n'. After Free Form finds the first row, it automatically locates all other rows. However, if Free Form does not find the first row, it rejects the entire column. You can prevent this problem by ensuring that the relative data search area is large enough to locate the first data item in the column.

11. Select the **Advanced** button on the Free Form Properties grid to display the Advanced Settings dialog.

12. Select the Free Form Column Properties tab and then the **Column Zone** check box to designate this field as a column header field. If you do not select this check box, the field will be processed as a normal Free Form field.

13. If you want column processing to flag any blank column rows as rejected, ensure that the **Flag Blank Fields** check box is selected. Blank fields are flagged with the "~" character in Designer and Completion will stop on each blank cell so you can enter data for the rejected cell. If you do not select this check box, column processing will not flag blank column rows as rejected. Blank rows will appear as blank fields in Designer and will be ignored in Completion.

Note: Free Form considers any field that does not match the regular expression to be blank. Therefore, a field does not actually have to be completely blank to be marked blank by Free Form.

14. In the **Minimum** and **Maximum** fields in the **Number of Rows** area, select the minimum and maximum number of rows within the column recognition zone. The **Minimum** value indicates the minimum number of rows allowed in a column. If there fewer rows found than the specified **Minimum** value, the entire column is rejected. The **Maximum** value indicates the row number at which to stop column processing. For example, if more rows are found

than specified by the Maximum value, only the maximum number of rows are processed. The additional rows are ignored and not output. Use of these parameters is optional and you can leave the values set at 0 (zero).

15. In the Identify Last Row By area, select the **Blank rows - minimum number** check box to terminate a column when one or more blank rows are found.

16. Select the **Stop Pattern** check box to automatically terminate a column when a desired data pattern is recognized. In the adjacent field, type the character pattern upon which column processing should stop when found.

17. Refer to the Locating the End of a Column section below for more information about setting the options in the Identify Last Row By area.

18. To indicate that the process should use the column keyword location on the previous form as a reference when searching for keywords and related data in this column, select the **Use previous image keyword coordinates** check box.

19. This option works only if your column is continued from a table on a previous form (i.e., the form is the second or third in a series, etc., and not the initial form) and you have created a blank keyword in the Free Form Properties grid. The search process uses the keyword in the column on the previous page to locate the column on the current page. However, you must have created a blank keyword record including the coordinates of the appropriate relative data search area.

20. To associate a field from the previous form with the current column, select the **Use Keyword Location from Previous Page** in the **If Column Keyword Not Found** area. Then, select a field on the previous form from the **Select Previous Page's Field** drop down list to associate it with the current column. Any field on the previous form that you have identified as a column zone displays in the drop down list.

21. Refer to the Defining Multi-Page Columns section below for more information about using this option.

22. Select **OK** to confirm your changes and return to the Free Form Properties dialog. Select **OK** again to return to the Field Properties, Recognition tab.

23. Repeat Steps 2 9 for each column on the image that you want Free Form to process.

24. After you have created all necessary fields and Recognition zones, select the **Table Zone** field type button on the Designer toolbar. Place the cross-hair on the data form to create a red table zone place-holder field.

25. Create a large Recognition zone on the image to act as the Table Recognition zone. This area must encompass all data, including all keywords, on the setup image.

26. It is critical that this area be equal to or larger than all other Recognition zone fields combined, such that all column fields physically reside within the Table Recognition zone. Free Form column processing will *not* work if any of the column image zones overlap or lie outside of the Table Recognition image zone.

27. Refer to the Defining Recognition for Table Zones portion of the Defining Recognition Zones section for more detailed information on setting up a table zone.

Forcing KFI Data Entry in Columns

You may want to require users to perform KFI (Key From Image) on Free Form column fields where Free Form Recognition might be poor. One example of such a field is a handprint field on a credit card application. When you force KFI for a field, this means that Completion always

displays the field as a blank field and the user must key it for each image. Free Form Recognition usually searches for a keyword and the location of the data associated with that keyword. When you use forced KFI with Free Form Recognition, Free Form rejects the field without searching for the data pattern. Thus, the keyword location alone forces KFI on the field.

Perform the following steps to force KFI data entry in a column field.

1. Open your job in Designer.

2. Perform steps 2 through 6 from the Creating Column Fields section above.

3. To search for only the keyword and not the data pattern, create a keyword as described in the Creating Keywords and Data Patterns section above, but *do not* create a relative data search area. This directs the user to what needs to be keyed and does not search for data.

4. Select the **Advanced** button on the Free Form Properties grid to display the Advanced Settings dialog.

5. Perform steps 9 and 10 from the Creating Column Fields section above.

6. Select the General tab on the Advanced Settings dialog and note three check-box options in the Directed KFI area. The **Set KFI from Keyword** and **Include Keyword in KFI** checkboxes are currently disabled because your free-form zone is a column zone.

7. Select the **Don't search for data** checkbox. When you select this check box, Completion displays the data clip as a directed clip relative to where the actual keyword is found. No Free Form data regular expression search is performed. The actual data clip never displays in Completion because the output is always blank and rejected.

8. Select **OK** to confirm your changes, close the Advanced Settings dialog, and return to the Free Form Properties dialog.

9. Perform steps 11 through 19 from the Creating Column Fields section above as needed.

Creating Multi-Line Column Titles

A multi-line column title contains text on more than one line, as shown in the following example.

Unit Amount	Number of Units	Total Amount
52.00	1	52.00
12.00	4	48.00
82.00	2	164.00
3.50	3	10.50

Creating multi-line column headers is very similar to creating single-line column headers. Perform the following steps to create a Free Form column header field for a multi-line column title.

1. Perform steps 1 through 6 from the Creating Column Fields section above.

2. Double-click on the column header field to open the Free Form Properties dialog.

3. Locate the multi-line keyword and recognition area in the image viewport.

4. Select the **Draw the Keyword to Search for** button. Cross-hair lines display for drawing the keyword zone.

5. Using your mouse, draw a zone around the keywords you want to use as a multi-line keyword. Upon releasing the mouse, a red border displays tightly around the keywords.

6. Select the **Draw the Data to Search for** button or **Draw the relative clip to search for data** button, depending on the type of clip you want to create, and draw the area for your data clip. In the grid portion of the Free Form Properties window, data fills in for the keyword you have just created.

7. When you draw a multi-line keyword zone, Free Form inserts a line feed separator \n between the two keywords. For example, the Unit Amount header in the first column of the example above would display as Unit\nAmount.

8. You can view the separator in the Keyword and Key Found columns of the grid. You can insert any number of \n separators to search for any number of lines of keywords. However, multiple \n separators in a row and \n separators at the beginning or end are not acceptable.

9. Perform steps 8 through 19 from the Creating Column Fields section above as needed to finish creating the column on your data form.

Defining Multi-Page Columns

If you have a multi-page application with columns that span more than one page, but column titles appear only on the first page, use the Use Keyword Location from Previous Page option. The example below shows a situation where you could use this option. The Use Keyword Location from Previous Page option allows you to specify the first data form and create links from the other data forms in the document.

Page 1:

Unit Amount	Number of Units	Total Amount
52.00	1	52.00
12.00	4	48.00
82.00	2	164.00
3.50	3	10.50

Page 2:

61.00	1	61.00
4.00	2	8.00
4.75	2	9.50
12.00	3	36.00

Perform the following steps to create a Free Form column field for a second page column without a title.

1. After creating all the data forms that will exist in the document in Designer, select the first form in the document in the Job Browser. The Form Properties dialog displays.

2. From the Edits tab, select the **Start of Document** check box. This designates the form as the beginning of the document for the job.

3. In the Job Browser, select the job name to display the Job Properties dialog.

4. Select the Links tab. Each form you have created appears in lists on both sides.

5. In the **Number** field, select 1.

6. In the **Form** area, select the first form of the document.

7. In the **Link to** area, select the second (subsequent) form for the document. This links the first form to the second. In the form area, the value in the Links to column should reflect the number of the form to which you created a link.

8. Continue this process for each form in the document. Remember that the form number in the **Number** field should correspond to the order of the form in the document. The final form in the document does not have to link to another form, and by default, displays as being linked to itself.

9. Create column header fields, fields for all the other rows in the columns, and table zones for the first data form as described in the Creating Column Fields section above.

10. Create column header fields, fields for all the other rows in the columns, and table zones for the second and following data forms as described in the Creating Column Fields section above, but do not set the Free Form properties as described there. Instead, set the Free Form properties for those data forms as described in the remaining steps below.

11. Double-click a column header field on the second data form and to display the Field Properties dialog, and select the Recognition tab.

12. Even though there is not a column header on the second image, create the same keyword for this column as you created for the associated column on the first data form.

13. On the Free Form Keyword Properties dialog, fill in the fields in the Relative Data Search Area (Pixels) area to create a relative data clip in the blank row at the top of the column.

14. Right-click on the next available empty row in the grid control to display a context menu for that row.

15. Select **Properties** from the context menu to display the Free Form Keyword Properties dialog.

16. Leave the keyword blank and select **Apply** accept the default data populated into the remaining fields from the first row. The Free Form Keyword Properties dialog closes, returning you to the Free Form Properties dialog.

17. Select the **Advanced** button from the Free Form Properties dialog, then select the **Column Zone** check box on the Free Form Column Properties tab to enable the column processing options.

18. Select the **Use Keyword Location from Previous Page** check box from the If Column Keyword Not Found area.

19. Select the associated column field name from the previous page from the **Select Previous Page's Field** drop down list. The field names on the drop-down list are the fields on the previous data form that you defined as Free Form column fields.

20. Select **OK** to confirm your changes and return to the Free Form Properties dialog.

21. Perform steps 11 through 19 as needed for each remaining data form.

Locating the End of a Column

You can indicate the point at which a column field ends by creating a keyword or blank row field following the column data. Perform the following steps to create a Free Form column field that indicates the end of a column.

1. Select the column header field for the column you want to end using a keyword or blank row field.

2. Create the keyword and relative data clip in the Free Form Properties grid.

3. Select the **Advanced** button to display the Advanced Settings dialog.

4. Select the **Column Zone** check box on the Free Form Column Properties tab to enable the column processing options.

5. Select the **Blank rows - minimum number** checkbox in the Identify Last Row By area to indicate that the column ends when a specific number of blank rows are encountered during the keyword search. Type the minimum number of blank rows you want to use to end the column into the **Blank rows - minimum number** field.

6. To indicate a column with characters present at the point of the column end, select the **Stop Pattern** check box from the Identify Last Row By area and type the pattern to locate in the keyword search.

7. Select **OK** to confirm your changes and return to the Free Form Properties dialog.

Testing Free Form Recognition

After you finish creating all the Free Form fields and columns in your application, you need to test Recognition to ensure you configured your application correctly. There are three testing methods available for Free Form - Manual, Auto, and Auto Stop.

Performing Manual Testing

Use the Manual testing option to test a single image without storing test results. If you do not select either (or both) of the Show OCR results and Show Keyword results options, no dialog boxes display the test results during the testing process. Whether or not you select those options, the first keyword in the grid found with data matching the regular expression in the relative clip area always display in the Key Found and Data Found columns of the Free Form Properties grid after a test. However, if you select Test Row from a keyword's context menu, the grid displays any keyword and data patterns found for that keyword row.

Free Form is a diagnostic tool you use to set up keywords and data searches for your application. If you select both the Set KFI from keyword and Include Keyword in KFI checkboxes on the Advanced Settings, General tab, results of the box union are not displayed. Instead, you see the results as they would display for any other Free Form field. Refer to the Performing KFI with Free Form section below for more information about using the Set KFI from keyword and Include Keyword in KFI checkboxes.

Perform the following steps to perform Manual testing on a Free Form field.

1. Double-click a field on the data form to display the Field Properties dialog, and select the Recognition tab.

2. On the Recognition tab, select the **Free Form Properties** button to display the Free Form Properties dialog.

3. Select the **Manual** radio button.

4. Select the **Show OCR results** check box to display a diagnostic dialog showing the recognition results when you perform manual testing of the current field using Free Form.

5. Select the **Show keyword results** check box to enable a series of diagnostic dialog boxes showing whether the keywords were found when you perform manual testing of the current field using Free Form.

6. Select the **Test** button to run a Free Form Recognition test. The **Test** button activates only after you have entered at least one keyword row on the grid. Manual testing only tests the current image for the BDF.

7. Select **OK** on the diagnostic messages that display. Depending upon which check boxes you selected for the test, you could see some or all of the following dialogs.

   - Incoming OCR Results, which includes all the data recognized on the image.

   - Looking for exclude keyword, which displays the excluded keywords. The recognition test first attempts to locate excluded keywords.

   - Found exclude keyword, which displays the excluded words found in the recognition data.

   - Looking for keyword, which searches for the keywords you specified.

   - Found keyword, which displays all valid data for the keywords found during the test.

   - Found valid data near keyword, which displays any valid data found.

   - Column Zone Results, which displays for any column keywords specified.

8. To test this field on additional images, select images from the drop-down list in the Browse bdfs area and repeat steps 4 through 8.

9. When you have completed your manual testing, select **OK** to exit the Free Form Properties dialog and return to Designer.

Performing Auto Testing

To test keyword recognition on all images associated with a BDF without diagnostic messages, use the Auto method of testing. Test results are stored in an RPT file located in the \FormWare\Tmp directory and are displayed at the end of the test process. You can also view them at any time by selecting the **Show Report** button.

Perform the following steps to perform Auto testing.

1. Double-click a field on the data form to display the Field Properties dialog, and select the Recognition tab.

2. On the Recognition tab, select the **Free Form Properties** button to display the Free Form Properties dialog.

3. Select the **Browse** button from the Browse bdf's area and select a BDF to test.

4. Select the **Auto** radio button. The **Show OCR Results** and **Show Keyword Results** check boxes are grayed out.

5. Select the **Test** button. Status messages for the recognition engine display as each image in the BDF is tested. Upon completion of the test, the Free Form Test Report dialog displays.

6. The Free Form Test report displays the following information.

   - The name of the BDF you tested.

   - The number of images in the BDF.

   - The number of images upon which data was located.

   - The percentage of keywords for which data was found.

   - The number of keywords not located.

   - The number of keywords for which no data was found.

Results of subsequent tests you run using the **Auto** option will continue to display in this dialog until you exceed five reports. The, the oldest RPT files are overwritten as new reports are generated.

While viewing information in the Free Form Report Results dialog, you can double-click on any report line to display the Batch Details dialog. The Batch Details dialog displays the following information.

- BDF name
- Number of images in the BDF
- Name of the image upon which a field was not located
- Name of the image upon which the field was located
- Name of the keyword that was located
- The data that was found for the keyword referenced in the previous column
- The confidence level for the keyword that was located
- The keyword that was not located
- The confidence level for the keyword not located
- Basic errors that occurred during processing of a particular image when Free Form recognition failed on that image. Errors can any of those listed in the following table.

Free Form Error	#	Error Text
IDS_FIND_FAILED	284	"Free Form: Find Failed"
IDS_ERR_INVALID_FIELDNO	286	"Free Form: Invalid Field Number"
IDS_ERR_CREATE_OCR_OBJ	287	"Free Form: Could not create OCR object"
IDS_ERR_CREATE_RULE_OBJ	288	"Free Form: Could not create RULE object"
IDS_ERR_MISSING_KEYWORD	289	"Free Form: Missing keyword"
IDS_ERR_MAKE_FUZZY	290	"Free Form: Could not make fuzzy expression"
IDS_ERR_EXCLUDE_KEYWORD	291	"Free Form: Could not exclude keyword"
IDS_ERR_EXCLUDE_DATA	292	"Free Form: Could not exclude data"
IDS_ERR_READ_FUZZY_GUESS_FILE	293	"Free Form: Could not read fuzzy guess file\n %s%s" (where %s%s is the path and file name)
IDS_ERR_READ_TEXT_BUCKET	294	"Free Form: Could not read text bucket"
IDS_ERR_GET_FIELD_PROPERTY	295	"Free Form: Could not get field property"
IDS_ERR_FIND_FAILED_LOW_CONF	296	"Free Form: Find failed low confidence"
IDS_ERR_NOT_A_FREE_FORM_FIELD	297	"Free Form: Not a Free Form field"
IDS_ERR_CREATE_REC_CLIENT_PROCESS	298	"Free Form: Could not create recognition client process"
IDS_ERR_COUND_NOT_GENERATE_BDF	299	"Free Form: Could not generate bdf"
IDS_ERR_UNSUPPORTED_RECOG_ENGINE	300	"Free Form: Unsupported recognition engine"
IDS_ERR_UPDATE_FORM	301	"Free Form: Could not update form"
IDS_ERR_CLOSE_FORM	302	"Free Form: Could not close form"
IDS_ERR_CLOSE_JOB	303	"Free Form: Could not close job"
IDS_ERR_MISMATCH_IN_REG_EXP	304	"Free Form: Error or mis-match in keyword regular expression"
IDS_ERR_COL_PROCESS_FAIL	307	"Free Form: Error Column Processing Failed"
IDS_ERR_KWFIND_FAIL	308	"Free Form: Keyword Find Failed"

Free Form Error	#	Error Text
IDS_ERR_OCR_OPEN_FAILED	309	"Free Form: Error Could not open Ocr File"
IDS_ERR_IMAGE_NOT_THERE	310	"Free Form: Image does not exist in specified path"

Performing Auto Stop Testing

To test all images in a BDF without generating stored test results, use the **Auto Stop** method. This Recognition tests all images in the current BDF, but Free Form pauses on any images where a keyword and matching data are not found and displays an error message indicating the reason for the pause. During this pause, you can train a new keyword, data search area, and relative clip area and then select **Continue** button to proceed with testing.

Perform the following steps to perform Auto Stop testing.

1. Double-click a field on the data form to display the Field Properties dialog, and select the Recognition tab.

2. On the Recognition tab, select the **Free Form Properties** button to display the Free Form Properties dialog.

3. Select the **Browse** button from the Browse bdf's area and select a BDF to test.Select a Free Form field and display the Field Properties dialog.

4. Select the **Auto Stop** check box. The **Show OCR Results** and **Show Keyword Results** check boxes are grayed out.

5. Select the **Test** button.

6. When testing pauses, review the reason for the pause and perform training as required. Then, select the **Continue** button to complete the testing.

You should also test your application at the form, job and workflow levels before putting it into production. Refer to the Testing section for more information about testing at the form, job and workflow levels.

Performing KFI with Free Form

There are two instances in which you may use KFI (key from image) with Free Form. In the first instance, if Free Form finds the keyword for a field but does not find the associated data pattern, it displays a keyword-relative clip to enable you to perform KFI for that field. In the second instance, you may want to key a particular Free Form field for each image. In that case, you can configure Free Form to capture a keyword and display directed data clip from which you can perform KFI for that field.

Performing Directed KFI when OCR Fails

KFI of an empty Free Form field where the keyword is found but the expected data pattern is not found is difficult if you must search for the data visually in the entire zone clip. When Free Form finds a keyword on an untrained form but does not find any data, it displays a smaller clip, facilitating faster KFI. Free Form uses the upper left corner of the keyword to display a the keyword-relative clip, truncated by the field zone. Directed KFI applies both to variants and to untrained forms. It does not apply to blank keywords, due to the absence of a relative clip.

Creating Directed Data Clips for KFI

In some instances, you may want to capture a keyword and display a directed data clip without applying OCR to the data. Therefore, you can turn off the Free Form Recognition search for the data pattern field by field. Once you turn off OCR for a given field, Free Form rejects the field and outputs blank results.

Perform the following steps to generate a small, directed data clip to be used for KFI.

1. Open your job and form within Designer.

2. Double-click a field on the data form to display the Field Properties dialog, and select the Recognition tab.

3. On the Recognition tab, select the **Free Form Properties** button to display the Free Form Properties dialog.

4. Using the scroll bar and the **Zoom-in** and **Zoom-out** buttons on the toolbar, locate the desired zone for the directed clip in the image display area.

5. If keywords already exist in the grid control on the Free Form Properties dialog, left-click anywhere in the next available empty row in the grid to enable the **Draw the Keyword to Search for** (key icon) button.

6. Select the **Draw the Keyword to Search for** button. Cross-hair lines display for drawing the keyword zone. Using your mouse, draw a zone around the keyword you want to create. Draw the zone tightly enough to *exclude* any other characters or black blobs within the drawn zone area. Upon releasing the mouse, a red border displays tightly around the keyword.

7. Select the **Draw the Data to Search for** button to draw the data zone around the desired data pattern, estimating how far to the right and down a zone might be. When you release the mouse, two borders display: blue is the data area and green is the relative data search area. Free Form uses the data pattern captured in the blue area to create a regular expression for the Free Form data search. In the grid portion of the Free Form Properties window, data will fill in for the keyword you have just created. The grid is populated based on the location of the keyword zone, the size of the data search area, and the type of data that you have included in the data search area.

8. Because you are using keywords to direct the user to areas on the image where no OCR is found, you should define an expected search area, also known as a directed clip, to search for the data associated with the keyword**.** The directed clip can be either the data relative clip or the box-union of the data-relative clip and the keyword clip. This area is defined in pixels as it relates to the location of the upper left corner of the first character in the keyword found in the keyword search. The coordinates of this area appear in the **X**, **Y**, **Width**, and **Height** columns of the Free Form grid. You are preparing a field for directed KFI when OCR finds no data, so avoid the **Draw the relative clip to search for data** button.

Note: You can opt to include the keyword within the directed clip in Completion. The default mode, however, is not to include the key word and to only display the data relative clip.

9. Select the **Advanced** button from the Free Form Properties dialog to display the Advanced Settings dialog.

10. Select the General tab on the Advanced Settings dialog to display the General tab.

11. The three options in the Directed KFI area affect Recognition and Completion as follows.

- **Set KFI from Keyword**: Selecting this checkbox enables the Don't search for data and Include Keyword in KFI checkboxes. Subsequently, three results are possible.

o   **Keyword is found**: In instances where the keyword is found for a given field, Completion displays a directed clip near or around the found keyword and displays a blank result. The directed clip is either the data relative clip or the box-union of the data-relative clip and the keyword clip, depending on whether you select the Include Keyword in KFI check box.

o   **Keyword is not found**: In instances where the keyword is not found for a given field, Recognition outputs a large blue clip and a blank result to Completion.

o   **No non-blank keywords found**: In instances where no non-blank keywords are found for a given field for which a blank keyword is defined, Completion displays a large blue clip and a blank result.

- **Don't search for data**: If you select the Don't search for data checkbox, Completion displays the data clip as a directed clip relative to the location of the keyword, if found. No Free Form data regular expression search is performed, and the output is always blank and rejected.

- **Include Keyword in KFI**: If you select the Include Keyword in KFI checkbox, Completion displays the directed clip as the box-union (clipped to the zone boundaries) of the keyword clip and the data relative clip.

12. Select **Set KFI from Keyword** and **Include Keyword in KFI** to configure your field for directed KFI.

13. Select **OK** to confirm your changes, close the Advanced Settings dialog, and return to the Free Form Properties dialog.

When Free Form processes a field configured as described above, Recognition OCRs the form and finds the keyword and the area where the data should exist. It fills the empty data field with a single space character and gives that space character the keyword-relative coordinates, truncated by the zone boundaries for the found keyword. Subsequently, Free Form uses the keyword to direct the user to the field where data needs to be keyed.

Validating Data Using a Lookup Table

You can validate found data by choosing a lookup table from the Advanced Settings dialog, General tab accessed by selecting the Advanced button on the Free Form Properties dialog. When testing occurs and Free Form finds data that matches the regular expression within the relative clip, it performs one more validation to see if the data exists in the defined lookup table. If the data is not there, Free Form rejects the found data.

Perform the following steps to define a lookup table to validate data.

1. Open your job and form within Designer.

2. Double-click a field on the data form to display the Field Properties dialog, and select the Recognition tab.

3. On the Recognition tab, select the **Free Form Properties** button to display the Free Form Properties dialog.

4. Select the **Advanced** button on the Free Form Properties dialog to display the Advanced Settings dialog, and select the General tab to display the General tab.

5. Select the **Browse** button from the Table values group to display the Open dialog. The Open dialog displays the contents of the \FormWare\Rules folder.

6. Select a table from the Rules folder that you want to use to validate found data for the current field.

7. Select **Open** to confirm your table choice, close the Open dialog, and return to the Advanced Settings dialog. The table name and path display in the Table field. Values contained in the table display in the Existing Table Values list.

8. To add a new value to the table, select **Add**. The Add Table Value dialog displays. Type a new value into the **Add Value** field. Select **OK** to confirm your changes, close the Add Table Value dialog, and return to the Advanced Settings dialog. The new table value displays in the Existing Table Values list.

9. To delete a table value, select a value, then select **Delete**. A message box asks if you are sure you want to delete the selected value. Select **Yes**. Free Form deletes the selected value from the table.

10. To edit a table value, select **Edit**. The Edit Table Value dialog displays. Edit the value in the **Edit Value** field. Select **OK** to confirm your changes, close the Edit Table Value dialog, and return to the Advanced Settings dialog. The edited table value displays in the Existing Table Values list.

11. To reset the table values you edited and remove the table name and path from the Table field, select **Reset Values**.

12. Select **OK** to confirm your changes, close the Advanced Settings dialog, and return to the Free Form Properties dialog.

Setting Advanced Recognition Properties

You can perform the following advanced recognition functions:

- Set properties for multiple fields at the same time.
- Create dictionaries.
- Create voting algorithms and voting DLLs.
- Set foreign language support.

Setting Properties for Multiple Fields

Within Designer you can change the properties for several fields at one time from Data Form context menu.

Note: You can also change properties for multiple fields on the Multi-Field Properties dialogs within Designer. See the Editing Properties for Multiple Fields section for more information.

To set values for multiple fields:

1. In Designer, open the job and form for which you want to set properties.

2. In the data form area on the Designer window, use your mouse to draw a "rubber band" around the fields for which you want to set properties.

3. Right-click on one of the fields.

4. From the menu that displays, select from the following options and set properties for all fields selected. (For more information on any of the dialogs that display, see the appropriate section in this guide.)

- o **Select Engine(s)** - Displays the Available Engines dialog. Select one or more recognition engines and select **OK**. To select more than one engine, press the [Ctrl] key while selecting the engines you want.
- o **Select Field Type** - Displays the Field Types dialog. Select a field type and then select **OK**.
- o **Select Point Size** - Displays the Select Point Size dialog. Type or select the desired point size for the text within the data fields and select **OK**.
- o **Select Chars per Inch** - Displays the Characters Per Inch dialog. Select the number of characters you want to allow per inch and select **OK**.
- o **Select Radio Button Group ID** - Displays the Radio Button Group ID dialog. Type the ID for the selected group of radio buttons and then select **OK**. This ID indicates to Designer that the radio buttons are a group and behave as one field (i.e., only one option within the group can be selected).
- o **Select Font** - Displays the Font dialog. Select the font and characteristics for the font and select **OK**.
- o **Select I-Form Options** - Displays the Field Group I-Form Options dialog. These options apply only to input, check box, radio button, list box, and combo box fields on an I-Form. The options define the edits, if any, that will execute against this field on the Internet browser side only.
- o **Select Column Grouping** – Displays the Column Grouping dialog where you assign members to groups.

Creating a Dictionary

A dictionary is a predefined list of field values that are expected as return data from a recognition engine. Dictionary files function similarly to look-up tables, but are accessed by the recognition engine, not the Completion operator. Within a dictionary file, you can set possible values that should be returned for the data field. The recognition engine compares its return values to those in the dictionary to determine low-confidence characters or to reject high-confidence character combinations not found in the dictionary.

For example, a dictionary defined for State fields contains the two-character abbreviation for each state. If the recognition engine is 95% confident that it has read "ZZ," but the dictionary contains no "ZZ" value for State, then recognition will reject the field.

You can define dictionaries on a form or field level, depending on the recognition engine you are using. All recognition engines support dictionaries at the field level, however CGK dictionaries must be prepared at the form level. You create dictionary files in the FormWare text editor. They are saved in the \FormWare\Lists folder with an .lst extension. Each value in the dictionary is on a separate line in the file.

Creating a Field-Level Dictionary

You can edit, compile, and assign dictionaries to fields on your data form. The InScript, InScript2, Nestor, RecoStar Professional, and Parascript recognition engines support field-level dictionaries. Before you use a dictionary, you must create a table from the dictionary values using the Table Preparation dialog.

Note: If you specify the InScript recognition engine for a field using a dictionary, you must write a rule to specify the user of the table. You do not need to specify a rule when using the Nestor or Parascript recognition engines.

To create a dictionary for the current field:

1. Double-click the field for which you want to create a dictionary. The Field Properties dialog displays.

2. Select the Recognition tab.

3. Select the **Dictionary Configuration** button. The Dictionary dialog displays.

4. In the **Dictionary Name** field, select **New**. The Enter Name dialog displays.

5. In the **Name** field, type a name for the new dictionary and select **OK**. The name displays with a .lst extension in on the Dictionary dialog.

6. Select the **Edit** button. The FormWare Editor displays.

7. Type the values for the dictionary. Each value in the dictionary must be on a separate line in the file. When you are finished, select **OK**.

8. Select the **Prepare** button. The FormWare Table Preparation dialog displays. You may also see a message asking you to select tables. Select **OK**.

9. In the **Lists to be Prepared** field, select the dictionary file you want to create for this field. If the correct table is not in the list, select the **Change Set of Lists** button to select the LST file you want. From the Select window, locate the list file and select the **Open** button.

10. In the **Tables to Generate** list, select one or more recognition engines.

11. Select **Set** to set the recognition engines for the table to be generated prior to generating the table.

12. Select the **Generate Tables** button to prepare a table with the dictionary values. FormWare translates the LST file in to the appropriate format required for each table type. For example, the Parascript dictionary and the Nestor dictionary might require different formats; however, both can be generated from the same .LST file. A message window displays after the generation process.

13. Select **OK** to exit the message window.

14. Select **OK** on the FormWare Table Preparation dialog to return to the Dictionary dialog.

15. Select **OK** to save your settings and return to the Field Properties dialog.

16. You can now use the assigned dictionary with the current field.

Assigning an Existing Dictionary to a Field

To assign an existing dictionary to a field:

1. Open a job within the Designer module and access the Field Properties, Recognition tab for a specific field.

2. Select the **Dictionary Configuration** button. The Dictionary dialog displays:

3. Select an existing dictionary from the **Dictionary Name** field or select the **Browse** button to locate an LST file.

4. Select the **Edit** button to display the dictionary in the FormWare text editor.

5. Make any changes necessary to the dictionary. Then save and close the file.

6. To remove the dictionary file from the field, select the file name in the **Dictionary Name** field, and select the **Delete** button. The file is not deleted from the \FormWare\Lists folder.

7. Select **OK** to return to the Field Properties, Recognition tab.

Creating a Form-Level Dictionary

The CGK recognition engine requires that you create a dictionary at the form level. The form-level dictionary is available to all of the fields on the data form and is comprised of the existing dictionaries assigned to the individual fields on the data form.

Follow these steps to create a dictionary for the current data form:

1. Assign field level dictionaries to individual fields on the data form by following the steps in the previous sections.

2. Display the Form Properties, Recognition tab.

3. Select the **Create Form Dictionary** button to initiate the CGK Dictionary Generation Utility and generate the form level dictionary.

4. Select **Yes** to generate the form dictionary.

Creating Voting Algorithms

Using the FormWare recognition process, you can run a series of recognition engines and *vote* on recognition results to determine which engine provides the most accurate results.

Voting increases the recognition accuracy rate and the overall confidence level of the recognition process. However, the voting process increases recognition processing time due to the use of multiple engines. You enable recognition voting and select recognition engines within the FormWare Designer module.

FormWare provides three voting algorithms and also allows you to create DLL files for custom voting. You can create voting algorithms in Designer.

To create a voting algorithm:

1. Open the job in Designer.

2. Display the Job Properties dialog and select the Recognition tab.

3. In the **Voting** area, select from the following check box options:

    o   **User defined DLL** - Indicates a user-defined Dynamic Link Library (DLL) will perform voting. This option is mutually exclusive from the other three. See the FormWare Programming Language Guide for steps on creating a DLL.

    o   **Select if match** - Accepts the character with the highest confidence when multiple recognition engines return the same character and the confidence level from both engines exceeds the confidence level specified on the field.

    o   **Select highest confidence** - Accepts the character with the highest confidence level when multiple recognition engines return different characters and the confidence level for both characters exceeds the confidence specified for the field.

    o   **Select highest spread** - Accepts the character that has the highest spread in confidence between the primary character and the alternate character when the primary character's confidence level exceeds the confidence level specified for the field. Select this option when you are using an engine that provides an alternate character.

4. Voting follows a hierarchy when you select more than one of the following voting algorithm options: Select if match, Select highest confidence, and Select highest spread. When you

check the Select if match option, the algorithm uses the matching character. If the Select if match option is not met, the algorithm applies the Select highest confidence criterion. If this is not met, the algorithm uses the Select highest spread criterion. If the characters returned from multiple engines fail all three checks, the character with the higher confidence is placed in the recognition results file.

Note: Voting algorithms only apply to handprint and machine print fields. These algorithms are not used on mark sense or bar code fields.

Creating a Voting DLL

Function calls are used in the voting DLL to perform various tasks against the batch being created. This section contains a list of the FormWare function calls used to create a voting DLL. After writing your program, compile your DLL program and place the DLL in the \FormWare\Programs folder. The FormWare-supported recognition engines are 32-bit engines and require that you compile the DLL using a 32-bit compiler.

Recognition voting is performed on each field. For this reason, you need to call the FWVote function once for each field. Include the FWVOTE.H file you create in your DLL program. This file should be located in the \FormWare\Other folder. To use your voting DLL, you need to enable recognition voting during application development..

When you use the FWVote Function, the program performs a voting algorithm and returns an array (that the program allocates and populates) of VOTING_DATA elements, whereby the primary character, its alternate character (if enabled), and its coordinates are written to the output file for this engine. Your program must also return (in the numVotingResultsPtr) the number of VOTING_DATA elements in the allocated array (votingResultsPtr). Notice that votingResultsPtr is a VOTING_DATA** so you will have to reference it as *votingResultsPtr (see the Voting_ Data Parameter section that follows). FWVote accepts seven parameters (five as input, two as output).

Voting_Data Parameter

The structure of the VOTING_DATA parameter is:

```
typedef struct _voting_data_
{
   char   primaryCharClass;    // The ASCII character
   long   primaryConfValue;    // It's confidence value (in percent)
   char   validAlternates;     // TRUE if the alternate characters are
  valid
   char   altCharClass;        // The ASCII character
   long   altConfValue;        // It's confidence value (in percent)
   long   xTl;                 // The X Top left coordinate
   long   yTl;                 // The Y Top Left coordinate
   long   width;               // The width of the character (in
  pixels)
   long   height;              // The height of the character in pixels
}  VOTING_DATA;
```

FWVote Syntax

The syntax for the FWVote function is as follows:

```
long FWVote(long numEngine1Chars, VOTING_DATA* engine1Data,
   long numEngine2Chars, VOTING_DATA* engine2Data,
   long userSpecifiedConfidenceValue,
```

```
long* numVotingResultsPtr, VOTING_DATA** votingResultsPtr);
```

FWVote Parameters

numEngine1Chars	Specifies the number of characters in the engine1Data parameter.
engine1Data	Points to an array of VOTING_ENGINE elements, which contains any recognition results generated by a previously-executed recognition engine for this field.
numEngine2Chars	Specifies the number of characters in the engine2Data parameter.
engine2Data	Points to an array of VOTING_DATA elements, which contains the recognition results that the current engine just generated.
userSpecifiedConfidenceValue	Contains the confidence level that characters must meet. Characters that do not meet the confidence level will be rejected.
numVotingResultsPtr	Contains the primary character, its alternate character (if enabled), and its coordinates that are written to the output file for this engine.
votingResultsPtr	Indicates the number of VOTING_DATA elements in the allocated array.

FWVote Return Value

Upon successful completion, the FWVote function should return a 0 (zero) value. If there was an error, the program must return a non-zero value. It is also important to note that under a successful completion scenario, you do not have to de-allocate the memory allocated for the votingResultsPtr array. The calling program will de-allocate the memory.

On an error condition, simply de-allocate any memory that you have allocated, NULL out the votingResultsPtr and return a non-zero value.

FWVote.dll File Sample

In the following file example, the FWVote function takes the results of the engine with the highest confidence value. If the characters from both engine1 and engine2 do not match, the confidence value is set to the Reject Confidence value of 254. Code has been added to handle a situation where one engine returns more characters than another. The extra characters are used, but they are given a Reject Confidence value of 254.

```
#include "fwvote.h"
#include <malloc.h>
long FWVote(long numEngine1Chars, VOTING_DATA* engine1Data,
long numEngine2Chars, VOTING_DATA* engine2Data,
long userSpecifiedConfidenceValue,
long* numVotingResultsPtr, VOTING_DATA** votingResultsPtr)
{
    int numMinChars;
    int numMaxChars;
VOTING_DATA* pNode;
    VOTING_DATA* pHighConfEngine;
// engine with the highest confidence
VOTING_DATA* pMaxCharEngine;
// engine with higher number of chars
    int i;
// Determine which engine had more chars.
```

```
    if (numEngine1Chars <= numEngine2Chars)
    {
    numMinChars = numEngine1Chars;
    numMaxChars = numEngine2Chars;
    pMaxCharEngine = engine2Data;
    }
    else
{
    numMinChars = numEngine2Chars;
    numMaxChars = numEngine1Chars;
    pMaxCharEngine = engine1Data;
}
    pNode = malloc(sizeof(VOTING_DATA) * numMaxChars);
    for(i = 0; i < numMinChars; i++)
    {
// Determine which engine had the highest confidence value
    if (engine1Data[i].primaryConfValue >= engine2Data[i].primaryConfValue)
        pHighConfEngine = engine1Data;
        else
        pHighConfEngine = engine2Data;
// Determine whether if the characters from both engines match
    if (engine1Data[i].primaryCharClass != engine2Data[i].primaryCharClass )
    pNode[i].primaryConfValue = REJECT_CHAR_CONFIDENCE_VALUE;
// reject this char
    else
    pNode[i].primaryConfValue = pHighConfEngine[i].primaryConfValue;
    pNode[i].primaryCharClass = pHighConfEngine[i].primaryCharClass;
pNode[i].validAlternates = pHighConfEngine[i].validAlternates;
pNode[i].altCharClass = pHighConfEngine[i].altCharClass;
    pNode[i].altConfValue = pHighConfEngine[i].altConfValue;
    pNode[i].xTl = pHighConfEngine[i].xTl;
    pNode[i].yTl = pHighConfEngine[i].yTl;
    pNode[i].width = pHighConfEngine[i].width;
    pNode[i].height = pHighConfEngine[i].height;
}
// If one engine had more chars use them, but give them reject confidence
value.
    for(i = numMinChars; i < numMaxChars; i++)
{
    pNode[i].primaryConfValue = REJECT_CHAR_CONFIDENCE_VALUE;
    pNode[i].primaryCharClass = pMaxCharEngine[i].primaryCharClass;
pNode[i].validAlternates =  pMaxCharEngine[i].validAlternates;
pNode[i].altCharClass = pMaxCharEngine[i].altCharClass;
    pNode[i].altConfValue = pMaxCharEngine[i].altConfValue;
    pNode[i].xTl = pMaxCharEngine[i].xTl;
    pNode[i].yTl = pMaxCharEngine[i].yTl;
    pNode[i].width = pMaxCharEngine[i].width;
    pNode[i].height = pMaxCharEngine[i].height;
    }
    *votingResultsPtr = pNode;
    *numVotingResultsPtr = (long)numMaxChars;
    return 0;
}
```

Setting Foreign Language Support for AEG and CGK

The AEG and CGK recognition engines support a number of foreign languages. The following table displays languages supported by the AEG and CGK engines:

Language	AEG	CGK
Austria		X
Belgium		X

Language	AEG	CGK
Czechoslovakia		X
France	X	X
Germany	X	X
Greece		X
Hungary		X
Italy	X	X
Netherlands	X	X
Poland		X
Russia		X
Scandinavia		X
Spain		X
Switzerland		X
Turkey		X
UK		X
USA	X	X

The AEG recognition engine provides an INI file for each of the languages it supports:

- AEGDUTCH.INI - Dutch
- AEGFRNCH.INI - French
- AEGGERMN.INI - German
- AEGITAL.INI - Italian
- AEGUS.INI. – US-English

These files contain parameters applying to the support of each particular language. The CGK recognition engine provides a single CGKClnt.ini file containing all of the language-specific parameters. All language-specific INI files are located in the \FormWare\INI\System folder.

When the AEG or CGK recognition engine receives a new job to process, it looks at the settings in the job INI file to determine the language type and then initializes to process the job in that language. It then processes all images in the job using the appropriate language.

Adding a Language to the Job INI File

Use the following instructions to apply the appropriate INI file parameter to use foreign language support. To edit your job INI file:

1. Open an existing job in Designer and display the Job Properties, General tab.

2. Select the **Edit .INI** button. This displays the INI file for the current job in the FormWare text editor.

3. Edit or add the following variable to this INI file:

   ```
   LANGUAGE=<languageName>
   ```

4. Where <languageName> = the appropriate language value. Valid values for the AEG recognition engine are:
   - DUTCH
   - ENGLISH_US
   - FRENCH

     o   GERMAN

     o   ITALIAN

5. Valid values for the CGK recognition engine are:

     o   AUSTRIA

     o   BELGIUM

     o   CZECHOSLOVAKIA

     o   FRANCE

     o   GERMANY

     o   HUNGARY

     o   ITALY

     o   POLAND

     o   RUSSIA

     o   SCANDINAVIA

     o   SPAIN

     o   SWITZERLAND

     o   TURKEY

     o   UK

     o   USA

6. Save your changes and exit the text editor.

Using a US Keyboard

To use foreign language support in the Designer module, type foreign characters (i.e., using a foreign character keyboard), or copy and paste characters from a program such as Character Map. Character Map is available on Windows NT.

The following steps provide an example of how to use the Character Map program to define a foreign language field. To define a German Handprint Alpha Field that has a capital U with an umlaut, do the following:

1. Open the Designer module and create an **Alpha Only** field on the data form and corresponding zone on the image.

2. Access the Character Map program on your workstation (typically within the Windows Accessories program group) and select the character Ü (with an umlaut).

3. Select the **Select** button to select this character.

4. Select **Copy** to place this character in your copy buffer. The character appears in the **Characters to Copy** field.

5. Return to the Designer module and display the Field Properties dialog for the **Alpha Only** field you defined.

6. Display the Field Properties, Selective Edits tab for the field.

7. Right-click on the **Include** field to display the context menu.

8. Select **Paste** to paste the character into the field:

9. Test the recognition on this field using the **Test** button on the Field Properties, Recognition tab.

Writing Rules to Increase Recognition

FormWare recognition provides high acceptance and accuracy rates and powerful recognition software due to its unique set of optimizing rules. These rules let you extract or parse relevant data, modify it, and validate it. Recognition rules use contextual information about the contents of data fields to maximize the field recognition results. During recognition, each field is evaluated, processed, and then optionally sent through the rule-based validation. The rule-based validation process increases the accuracy of accepted data, minimizes substitution, and determines which fields require reject repair. Recognition rules let you perform *fuzzy lookups* that use wildcards or alternate characters to match or substitute field data. You can write customized rules or requirements for each field.

Rules are a method of using contextual information about the contents of data fields to maximize field recognition rates. Character accuracy is often not as important as field accuracy and the cost of errors in those fields. Sometimes, when data is densely packed together, the system must first extract groups of characters into fields. Each field must have specialized rules and lookups to determine its acceptability after recognition.

FormWare provides a rules language that enables you to write individual field rules for data acceptance, and the option of writing VBA edits for any field. For example, if your data form has an alphanumeric field that must start with an alphabetic character and the letter I is being recognized as the number 1, you can write a rule to substitute all number 1s in the first position with the letter I. In this way, rules can greatly increase accuracy and acceptance rates.

Understanding Recognition Rules

Rules perform three primary functions:

- **Extraction** - Specifies data to be extracted from all characters recognized in a field.
- **Modification** - Formats or changes data.
- **Verification** - Enforces constraints on field values and determines whether field data should be accepted or rejected.

A rule file is an ASCII file containing data acceptance procedures written in a fourth-generation rule language created by FormWare. The rules in the rule file are written or specified in the Designer module. Rule files work with Form Definition Files (FDF) that contain the specifications the FormWare programs need to process batches of a specific form type. The FDF file controls all form processing and recognition.

The FDF file references the rule file for acceptance requirements. A single FDF file can only reference one rule file, but several FDF files can reference the same rule file. This is useful in applications that process many different types of forms that warrant separate FDF files, but have enough similarities to reference the same rule file. Additionally, a rule file can include other rule files using "# include <filename>".

The FDF file references a single master rule file, and, for each field being processed, the name of the rule that contains processing procedures for the field. The master rule file references a set of other rule files that comprise all the rules for all the fields.

Some of the field rule files reference lookup tables. Lookup tables and cross-lookup tables are separate ASCII files that are used to manipulate, accept, or reject field data. Lookup tables are used to reject a field if the field value does not match a corresponding value in the lookup table. If the field value does match a corresponding value in the lookup table, you can accept the field as is, or substitute the field value with a new value. A cross-lookup table contains a list of multiple

entries. Cross-lookup tables are used to reject a field if other related fields in the lookup-table do not match, such as city names, area codes, and zip codes.

Improving Recognition Using Rules

Rules also improve recognition by enabling the following features:

- **Parsing** - Extracts a portion of the recognized data. For example, you may want to remove hyphens from phone numbers.
- **Fuzzy lookups** - Uses alternate character results when the best guess results do not match any entry in the dictionary.
- **Filtering** - Extracts garbage or unnecessary characters. For example, you might not want to keep hyphens that are sometimes included in telephone numbers. Filtering removes the hyphens and saves only the numbers.
- **Coercion** - Changes recognized characters to the field-appropriate characters. For example, in a numeric field a character recognized as a B is substituted with an 8 because it is the closest, most appropriate character for the field.
- **PerfectPost** - Provides a lookup table for addresses. PerfectPost is a recognition engine that provides a table of known street, city, state, and zip code addresses.

To add rules to a form:

1. Display the Form Properties dialog for the form.

2. Select the Edits tab.

3. In the **Rule file** field, select the **Browse** button to locate the rule file.

4. Select the **Rules** button to edit the rule file.

5. On the data form, select the field for which you want to write a rule and display the Field Properties dialog.

6. Select the Selective Edits tab.

7. In the **Rule Name** field, select the **Browse** button. The Available Rules dialog displays the rules that exist in the rule file you selected. Double-click on a rule name to add it to the **Rule Name** field. Select the **Delete** button to remove the rule.

8. Select the **Rules** button to edit the rule. If you do not have a rule specified for this field, but do have a rule specified for the form (on the Form Properties, Edit tab), you can edit the entire rule file. This button is only active when a rule is specified in **Rule Name** field on the Selective Edits tab.

For detailed information about specific rules and how to write them, or for information on creating a rule file see the FormWare Programming Reference Guide.

Testing Recognition

Testing the recognition properties on different levels and with other processes is important because it verifies your production environment recognition results. Recognition is just one of a number of processes that you should test at the field, form, and job level.

When testing recognition, work from the field level to the form level, and then to the job level. Testing on the field level applies only the field recognition properties, while testing on the form

level applies the field and form recognition properties, and testing on the job level applies all recognition properties set within the job. By testing on these different levels, you will better be able to isolate problems.

When testing recognition properties, you conduct either discrete or comprehensive testing. Discrete testing tests a single process allowing you to isolate potential problems before moving on to a comprehensive test. Comprehensive testing combines many processes, allowing you to see how they impact Recognition. Additionally, discrete testing is conducted only at the field and form levels and applies to a single image, while comprehensive testing applies to all levels and a batch of images.

Performing Discrete Recognition Testing

To conduct a discrete test at the field level, complete these steps. You can also perform a discrete test on the form level.

1. Open the form in the Designer module.

2. Select the field you want to test and display the Field Properties, Recognition tab.

3. Verify that the settings on this tab are correct, including the **Recognition Type** and **Recognition Engine** field values.

4. Select the **Test** button to test recognition properties set for the current field. The selected recognition engine(s) runs against the current field. If the test is successful, the recognition results display in the field on the data form. When the engine returns question marks instead of characters, it indicates that the character(s) fell below the specified confidence level for the field.

5. Change recognition properties and continue to test them until optimal recognition results are achieved.

If you are testing with an image that requires form background removal, you must run Image Enhancement prior to recognition. To enable this feature, select the **Enhance Before Recognition Tes**t check box on the Form Properties, Enhancement tab before conducting a recognition test.

Performing Comprehensive Recognition Testing

Use the following steps to conduct a comprehensive Recognition test, or to test Recognition with other processes. The following steps perform a comprehensive test on the form level, although comprehensive testing can also be performed on the field and job levels.

1. Open a form within a job in the Designer module.

2. Display the Form Properties, Test tab.

3. Select the **Recognition** check box and the check boxes for any other processes you want to test.

4. Select the **Browse** button to display the Browse Files dialog and select a BDF file to test against the data form.

5. Select the **Test** button to run a comprehensive test on the current data form. Status dialogs display as each process you selected is performed on the data form. A comprehensive test on the form level applies properties set for all of the processes on the form and field levels. Recognition results display in the Designer module or in a modified Completion window (see

below), providing you with diagnostic information that enables you to determine whether to make adjustments.

6. To view recognition test results in Completion, select **View Tests in Completion** button on the Designer window. Selecting this option enables you to use the Completion module as a diagnostic tool for optimizing recognition results. See this example of the results:

7. The red boxes around the characters represent the segmentation of the characters. The field names display, followed by the field confidence and recognition results for the field. The field confidence is comprised of an average of the confidence levels returned for each character in the field. If the field is rejected, the field name and confidence display in red instead of black. If this were a field-level test only the selected field would display in the Completion window.

8. To view the recognition results file information for each character returned, display the drop-down menu for each field, as follows:

- **CHR** - The character returned by the recognition engine.
- **PCT** - The percentage of confidence for the returned character.
- **XPOS** - The x-coordinate location of the character.
- **YPOS** - The y-coordinate location of the character.
- **WID** - The width of the character in pixels.
- **HGT** - The height of the character in pixels.

Note: If you are testing your setup image against Full Text recognition, deselect the **View Tests in Completion** check box. The Full Text recognition component (FWFTClnt) converts the image and exports it to the file location you indicate in the **Export Path** field of the Full Text Properties dialog. From there you can review the recognition results using the application related to the conversion.

Performing a Field or Form-Level Recognition Test

Within Designer, you can quickly test recognition on one or more fields on a form. This type of recognition test also enables you to view OCR data for the field directly on the Designer window.

You can run this test for a single field (field-level test) or all fields on the data form (form-level test).

- Perform field-level recognition testing on the Field Properties, Recognition tab.
- Perform form-level recognition testing on the Form Properties, Recognition tab.

Field-level testing executes just the recognition process for the current field against the setup image. Form-level testing executes just the recognition process for all data fields on the current form. This is referred to as discrete testing.

You can also test multiple processes for a single field or all fields on the form against a batch of images from the Field Properties, Test tab or Form Properties, Test tab respectively. This is referred to as comprehensive testing. Use comprehensive testing to see how other processes, such as image enhancement, affect recognition results and properties.

To perform a discrete form-level recognition test within Designer:

1. Within Designer, create a data form with at least one recognition field and a corresponding recognition zone.

2. Set the appropriate field properties for each field to enable recognition.

3. Select **Edit > System Defaults** to display the System Defaults dialog.

4. Select the General tab and check the Display OCR result data after test option.

5. Select **Apply** and then **OK** to accept your changes.

6. Display the **Form Properties, Test** tab and verify that only the Recognition process is selected for the test.

7. Display the Form Properties, Recognition tab.

8. In the **Alternate characters** field, select the number of alternate characters you want the recognition engine to provide for characters within the field. The limitations of the recognition engine selected for each field determine the maximum number of alternate characters that can be retrieved.

9. Select the **Test** button. This runs the recognition engine(s) you specified for the field in the Recognition engines field on the **Field Properties, Recognition** tab. If recognition is successful, results will display in the field(s) on the data form in Designer.

Note: You can also perform this function for a single field on the form on the Field Properties, Recognition tab, Test button. To do this, you must first select the field you want to test on the data form and then display the Field Properties dialog for that field.

10. Review the OCR results for the field that display beneath the Job Browser on the Designer window.
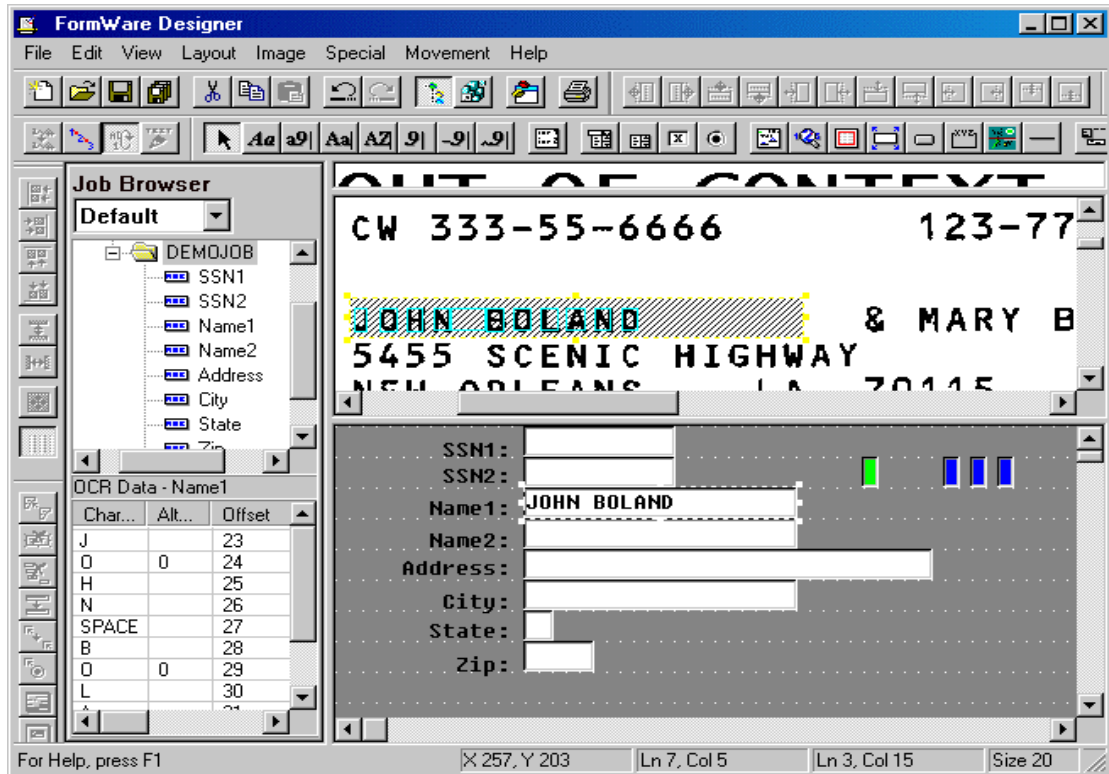
11. The highlighted zone within the Image Viewport is synchronized with the field selected on the data form. The corresponding OCR data results for this field display beneath the Job Browser. Each recognized character within the zone that is equal to or above the field confidence level is contained within a colored character clip rectangle. You can change this color in the Zone field of the Job Properties, General tab. When the recognition engine recognizes a character that falls below the field confidence level, the clip rectangle for that character will display in red.

12. Here is an expanded view of the OCR data results area:

OCR Data - Name1

Character	Alternates	Offset	Confidence	X Position	Y Position	Width	Height
J		23	100	161	215	7	13
O	O	24	100	173	215	9	13
H		25	100	186	215	8	13
N		26	100	199	215	8	12
B		28	100	224	215	9	13
O	O	29	100	237	215	9	12
L		30	100	250	215	9	12
A		31	100	263	215	10	12
N		32	100	276	215	8	12
D		33	100	289	215	8	12

- **Character** – Displays the best guess character recognized in this character position for the field by the recognition engine.

- **Alternates** – Displays the alternate characters recognized in this character position for the field by the recognition engine. The number of alternates available in this column for each character is determined by: the ability of the recognition engine selected for the field to produce such alternate character values and the value you set for the form in the Alternate characters field on the Form Properties, Recognition tab.

- **Offset** – Displays the position within the recognition results file where the recognition engine will begin to read data for this character. This is represented as a numeric value in pixels.
- **Confidence** – Displays the numeric percentage value that indicates how confident the recognition engine is of the value in the Character column. If this confidence value falls below the confidence value set for the field, that percentage value displays here and the entire row for this character displays in red.
- **X Position** – Displays the horizontal position, in pixels, of this character on the image. X and Y positions for each character intersect providing the precise location of this character on the image.
- **Y Position** – Displays the vertical position, in pixels, of this character on the image. X and Y positions for each character intersect providing the precise location of this character on the image.
- **Width** – Displays the width, in pixels, of the clip rectangle for this character.
- **Height** – Displays the height, in pixels, of the clip rectangle for this character.

13. Press **[Ctrl-Left Click]** to toggle on the image magnifier. (You can toggle off the magnifier in the same way.)
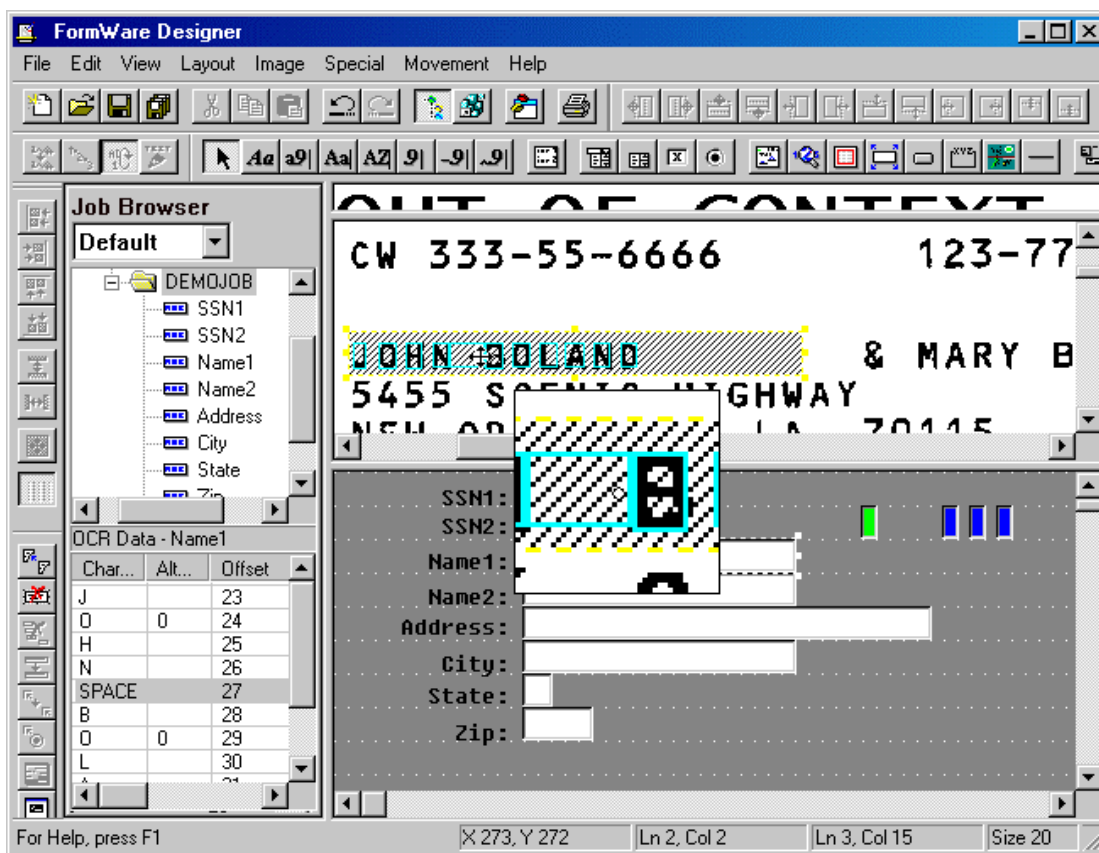
14. Once you display the image magnifier, press **[Ctrl-Right Click]** to toggle among the different magnifier sizes. Use the magnifier to gain better control of moving and resizing field zones to improve recognition.

15. Once you have your settings fine-tuned, select **Save All** on the toolbar to save your work before you exit.

Using the Zone Viewer

When you create a data form with many fields, the recognition properties of one or more fields may be set incorrectly. To help determine if the recognition is set correctly, use the **Zone View** option. The Zone Viewer displays a map of all of the recognition zones on your data form with each type in a different color.
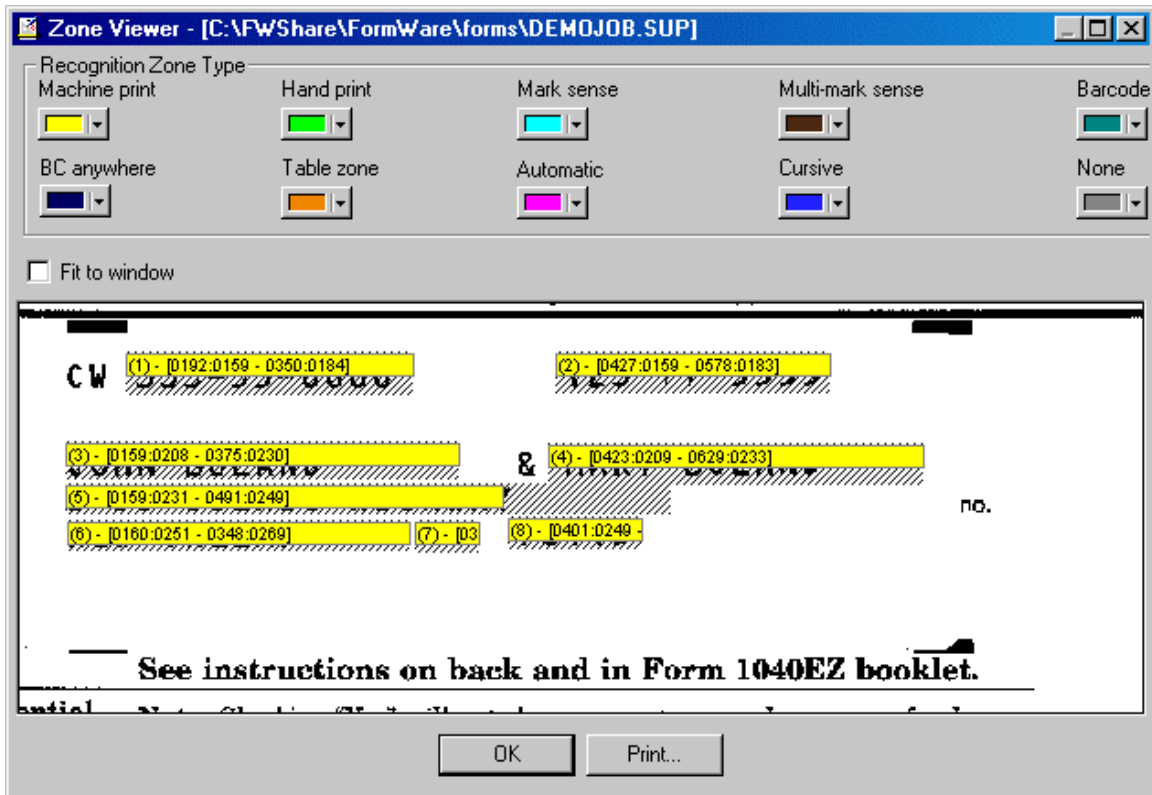
To view the entire data form, select the **Fit to Window** check box. To change the colors of a recognition type, click the down arrow next to the color to display the Windows color palette. The colors that you specify in the Zone Viewer are also visible in the Image Viewport so you can quickly move from field to field to check recognition types.

Generating a Recognition Results File

Recognition results (OCR) files provide detailed information about each field collected during the recognition process. OCR files are binary files that can only be viewed using a special FormWare Export utility. These files are generated as the result of a JobFlow recognition process run over a batch of images. Before you can generate an OCR results file, you will need:

- A job in Designer with at least one form containing fields zoned for recognition.
- A group of images located in the \FormWare\Images folder.
- A BDF that contains a pointer to your job and images in the \FormWare\Files folder.
- A JobFlow that contains at least one recognition task.

To run a JobFlow recognition task and generate OCR recognition results:

1. Open Administration and display the contents of the Files folder.

2. Highlight the BDF in the Contents area that contains the pointer to the job and images you want to process.

3. From the menu bar, select **Process > JobFlow** to display the JobFlow dialog.
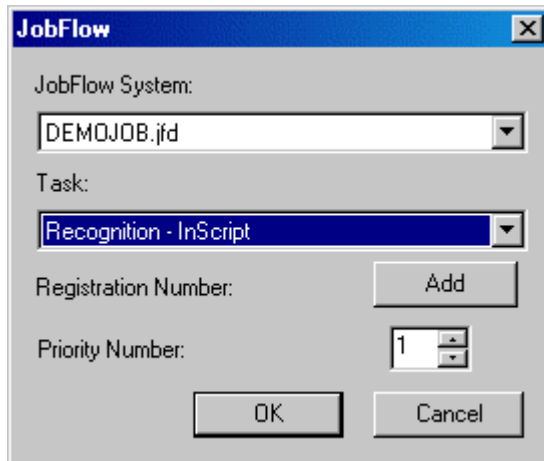
4. In the **JobFlow System** field, select the name of the JobFlow to process.

5. In the **Task** field, select the first recognition task within the JobFlow.

6. Select **OK** and the following dialog displays.
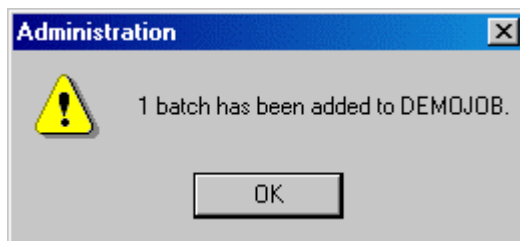
7. Select **OK** to return to Administration.

8. Select the **JobFlow Client** button on the toolbar. The JobFlow Client dialog displays.
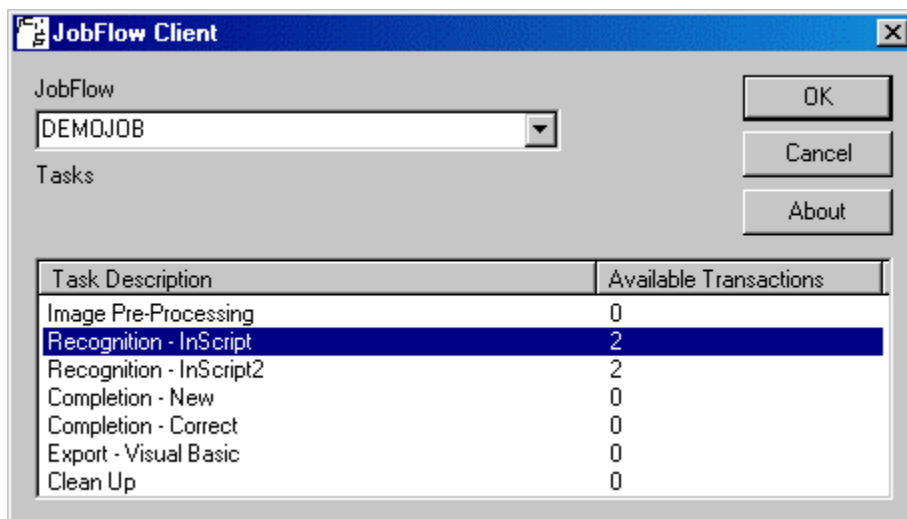
9. In the **JobFlow** field, select the name of the JobFlow to process.

10. In the **Tasks** field, highlight the first recognition process in the list that contains available transactions.

11. Select **OK** to process the JobFlow recognition task you selected. A number of dialogs display the progress of the recognition task processing.
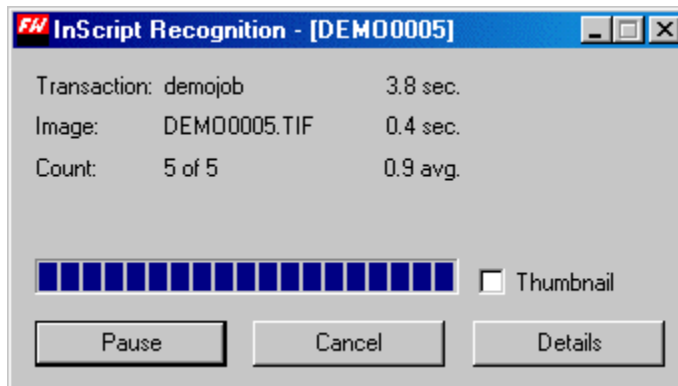
12. Select **Cancel** at the Unable to Continue dialog and return to Administration.

13. Within Administration, display the contents of the Images folder.
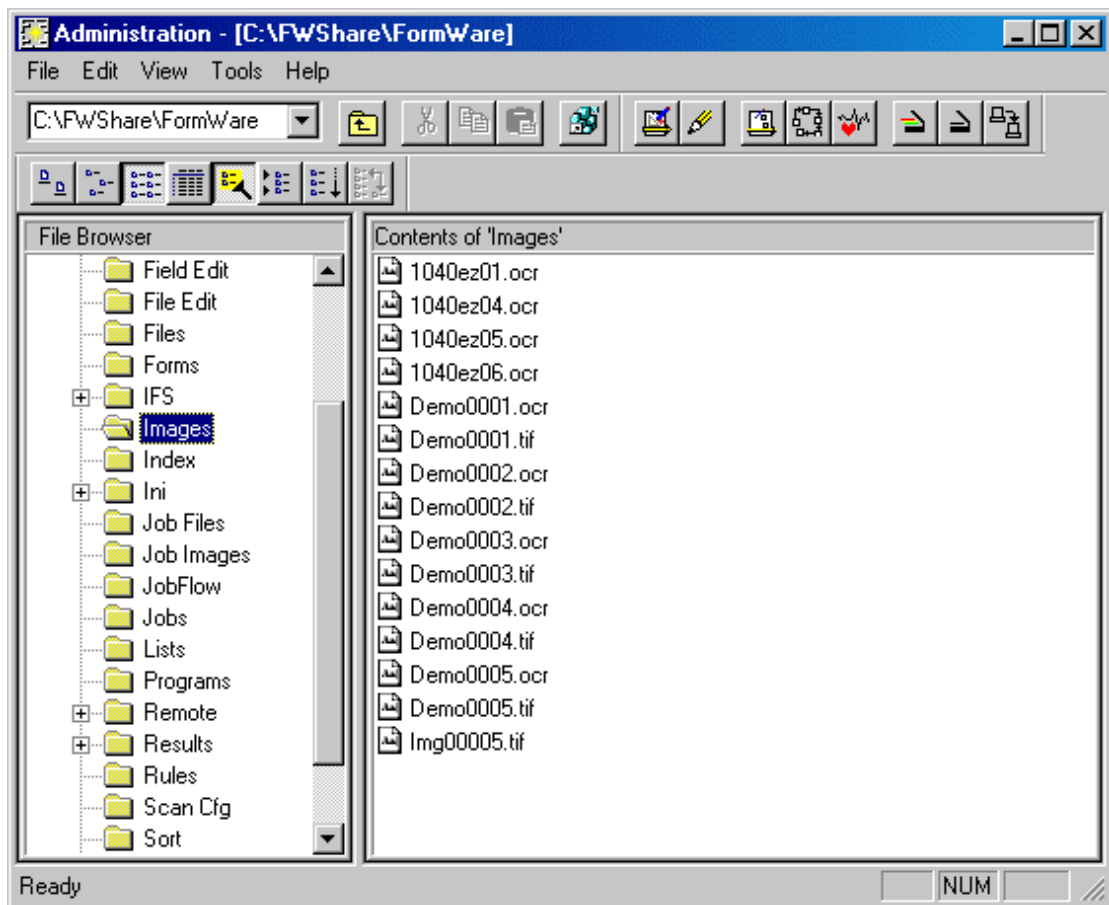
14. Double-click on an OCR file. This runs the FormWare Export utility and the OCR results display in a text viewer. The results data are presented only for your information and cannot be manipulated.

```
Editor - [C:\FWShare\FormWare\tmp\viewdata.txt]
File  Edit  View  Help

Ocrdata: 1040ez01.ocr
OFS     CHR   PCT   XPOS   YPOS   WID   HGT
====    ===   ===   ====   ====   ===   ===
0001    'M'   099   0118   0215   015   021
0002    'A'   099   0138   0216   014   020
0003    'R'   099   0159   0216   013   020
0004    'Y'   099   0178   0216   014   020
0005    'A'   099   0198   0216   014   020
0006    'N'   099   0218   0216   014   020
0007    'N'   099   0238   0216   013   020
0008    ' '   100   0000   0000   000   000
0009    'W'   099   0277   0216   014   020
0010    'O'   099   0297   0216   014   021
0011    'O'   098   0316   0217   014   020
0012    'D'   096   0338   0217   012   020
0013    'S'   099   0357   0217   013   020
0014    ' '   100   0000   0000   000   000

Ready                          Ln 1, Col 1        NUM
```

15. Results data includes:

- **OFS** - The offset in the OCR file, where the recognized image is located.
- **CHR** - The character recognized at the specified offset.
- **PCT** - The character confidence level.
- **XPOS** - The x-coordinate of the position on the image where the character is found.
- **YPOS** - The y-coordinate of the position on the image where the character is found.
- **WID** - The width in pixels of the character bitmap.
- **HGT** - The height in pixels of the character bitmap.

Setting the Data Form Display for Completion

The FormWare Completion module provides the tools necessary to perform post-recognition processing. The post-recognition process occurs after scanning, image pre-processing, form identification, form removal, and recognition are complete. This module includes powerful recognition reject repair, image data entry, key verification, and data validation. In addition to the built-in features of this module, you can use application-specific edits to remove erroneous data, perform table lookups, and perform mathematical calculations.

In Designer, you must first set up the Completion view for your operators. This includes:

Changing the Reject Ticker Tape Area

If the characters displayed in the reject ticker tape are too small, follow these steps to change the height of the characters or the space between the characters:

1. From Designer, open a job and select a form within the job.

2. Display the Form Properties, Image tab for the form.

3. In the **Reject Height** field, select the height (in pixels) of the reject characters displayed in the ticker tape during a Completion session. Notice the height of the reject characters in the Designer window ticker tape changes as you change this value.

4. In the **Reject Fill** field, select the space (in pixels) between reject characters displayed in the ticker tape during a Completion session. Changes to the value do not change the appearance of the reject ticker tape characters within Designer; they are only visible during the Reject Repair mode of Completion.

5. Select the **Apply** button to apply your changes on this tab.

6. Select the **Save All** button on the Designer window toolbar to save your changes.

Setting the Viewport Origin for Key from Image

When you design a Key From Image (KFI) application, you use the image viewport, but you do not need to define recognition zones. This enables a KFI operator to see only the area that he is working on clearly within the viewport and reduce the time it takes to manually type the data.

Defining the Viewport Origin for a Field

To create a KFI application and set the viewport origin for fields:

1. Open the job in Designer.

2. Select the KFI form within the Job Browser and display the Form Properties, Image tab for the form.

3. In the **View** area, select the **Image Only** check box.

4. Select the field on the data form for which you want to set the viewport origin.

5. Use the right mouse button to move the setup image within the viewport so that the desired area displays.

6. Select the **Set Viewport Origin** button on the toolbar. This sets the viewport for the selected field.

7. Select the **Save All** button to save your changes.

Copying the Viewport Origin

Follow these steps to copy the viewport origin for any field marked as Image Only, Color Zone, or Framed Zone in the View area on the Form Properties, Image tab.

1. On the data form in Designer, select a field from which you want to copy the viewport origin.

2. With this first field selected, press the [Shift] key and then select the field(s) to which you want to copy the viewport origin.

3. Release the mouse and select the **Copy Viewport Origin** button on the Image toolbar. This copies the viewport origin from the first field selected to all other selected fields.

4. Select the **Save All** button to save your changes.

Setting the Recognition Zone View

After you finalize changes to the Image Pre-Processing properties for the setup image, and before you create fields, labels, and recognition zones, set the recognition zone **View** option. There are multiple options for displaying recognition zones in the viewport of the Completion module. The

option you select determines how the data form and image are displayed within Completion. The Movement menu and Fine Zone Adjustment toolbar on the Designer window enable you to move and resize the recognition zones on the data form.

To set the recognition zone view:

1. In Designer, select a form within a job

2. Display the Form Properties, Image tab for the form.

3. Select one of the following options in the View area for either Recognition or KFI applications:

    o **Color Zone** - Displays the image in the viewport and highlights the recognition zone with the color designated on the Job Properties, General tab.

    o **Masked Zone** - Masks the image and displays only the recognition zone.

    o **Framed Zone** - Displays the image and places a thin red line around the recognition zone.

4. Select the field for which you want to adjust the zone.

5. Use the following options on the Fine Zone Adjustment toolbar to move and resize the recognition zones on the data form.

    o **Zone Left, Right, Up, Down** - Moves the selected recognition zone one pixel to the left, right, up, or down.

    o **Zone Increase, Left, Right, Top, Bottom** - Increases the right or left side, or top or bottom of the selected recognition zone by one pixel.

    o **Zone Decrease Left, Right, Top, Bottom** - Decreases the right or left side, or top or bottom of the selected recognition zone by one pixel.

6. Select the **Save All** button to save your changes.

Modifying the Keying Order

By default, FormWare assigns a keying order based on the order in which you create the data fields. If you do not want to present the fields to the operator in this order, use the Keying Order option to renumber the fields on the form.

Note: The Keying Order option changes the field number, not just its keying order.

1. Open the job in Designer and select the form you want to modify.

2. Select the **Keying Order** button on the Special toolbar. The current keying order displays on the data form. The labels are assigned numbers also. While the cursor does not locate to a label during a Completion session, the label briefly becomes active.

3. Select the field you want to be first in the keying order. Its number changes to one. The keying number increments and the cursor displays the number that will be assigned to the next field. All subsequent fields also renumber.

4. Select each field in order. The number displayed on each field changes to reflect the new keying order.

5. When finished, select the **Keying Order** button again to turn it off.

6. Select **OK** on the confirmation window to save your changes or **Cancel** to retain the former order.

You can also change the keying order of a field starting with any field number.

1. Select the **Keying Order** button.

2. Hold down the [Shift] key and select the field that has the number preceding the one you want to change. For example, to change field 15, press [Shift] and select field number 14.

3. Continue clicking on the fields until they are in the correct order.

4. When you are finished, **Save** your changes, and click the **Keying Order** button to toggle it off.

5. Once you begin renumbering, you can reset the number and start over by double-clicking on any field or label. This sets the keying number back to one.

Adding Snippets to Your Application

A snippet is a portion of the image that is displayed on the data form. Snippets are not associated with any label or field. Use snippet for reject repair or with KFI applications. You can also create a snippet-only application, where just the fields and the snippets are visible, or use snippets with recognition zones.

Creating Image Snippets

To create an image snippet:

1. Open the job in Designer.

2. Select a form that has a setup image, data fields, and recognition zones.

3. Select the **Image Snippet** button on the Field Types toolbar.

4. Click on the data form and drag the mouse to create the snippet zone. Unlike the data fields, snippets can be larger than a single grid row. Do not draw the zone over any existing fields on the data form or they will be covered by the snippet.

5. Release the mouse button and the new snippet displays on the data form in a grayed- out mode.

6. Move the setup image within the image viewport so that the information you want to associate with this snippet displays in the viewport.

7. In the image viewport, draw a zone on the setup image to create the snippet zone. The image snippet now displays on the data form.

8. Select **Save All** to save your changes.

Creating a Snippet Application

An image snippet application uses both an image viewport and image snippets. Follow these steps to create an image snippet application:

1. Create a new job in Designer.

2. When the Job Defaults dialog displays for the new job, select the **Snippet** check box in the Form Layout area. The resulting Designer window displays a small image viewport, which you can resize. The in-context viewport does not display during a Completion session for snippet applications.

3. Add a setup image on the Form Properties, Image tab.

4. Create data fields on the data form and draw recognition zones for them. Leave room between the fields on the data form for the image snippets.

5. Select the **Image Snippet** button on the Field Types toolbar.

6. Draw the snippet zone on the data form. Unlike the data fields, snippets can be larger than a single grid row. The new snippet displays on the data form in a grayed-out mode.

7. Move the setup image within the viewport so that the desired information on the image displays in the viewport.

8. Select the snippet on the data form and draw the snippet zone on the setup image in the image viewport. The image snippet now displays on the data form.

9. Repeat the steps above as necessary to create snippets for the remaining fields on the data form.

Creating Dynamic Snippets

Dynamic snippets are generated automatically based on the field recognition zone. Using dynamic snippets eliminates many of the previous snippet setup steps. During the Completion process, dynamic snippets display on the data form.

To use dynamic snippets in your application:

1. Open an existing job in Designer.

2. Select the form that contains a setup image, data fields, and recognition zones.

Note: All fields and labels must be left justified when you use dynamic snippets.

3. Open the Job Properties, Display tab.

4. Select the Display Field Snippets and Hide Image Display check boxes.

5. Select the **Save All** button to save your changes.

Reviewing Rule Recognition Statistics

Using FormWare, you can generate recognition statistics that will assist you in optimizing application results and monitoring production-level activity. Optimizing your results to get the most from recognition uses a test suite of images. Results are taken from the images and then compared against answer files to determine accuracy and acceptance. Generating reports after you are in production can ensure that your environment has not changed.

Note: Recognition statistics are not applicable for full text recognition.

Setting Statistics Job Properties

The options on the Job Properties, Statistics tab control the display and creation of operator and recognition statistics. All statistics files are stored in the \FormWare\Stats folder. To enable statistics:

1. Open the job in Designer.

2. Display the Job Properties, Statistics tab.

3.   Select the **Operator** check box to write statistics to the Operator Statistics file whenever a batch is terminated or interrupted. If you do not check this option, no statistical information is saved.

4.   Select the **Location only** check box to not update the net keystrokes if the operator backs up to a previous field or **record.**

5.   Select the **Save All** button to save your changes.

Generating Recognition Statistics

To measure the performance of recognition for an application, you can obtain statistical information about the recognition performance of an application and measure the acceptance and accuracy of the recognition on one data form type at a time. The Stats procedure automates the measurement of recognition performance during the development process. It is not intended for use in a production environment. In combination with a well-selected test suite of images and files of expected results (answer files), it helps predict how well a particular application is likely to work in production.

The Stats procedure measures statistics on fields within a data form, rather than individual characters. It detects patterns of errors, targets specific areas needing improvement, and ensures that changes to a Form Definition file (FDF) have no adverse effects on character recognition.

Using the Stats procedure is a cyclical process that includes running recognition engines on a test suite of images, measuring performance, adjusting the FDF and rules, and then repeating the steps again. It enables you to quickly and accurately refine registration, zone sizing and placement, recognition parameters, and validation rules.

The primary goal of generating statistics is to measure acceptance and accuracy of recognition engines. Acceptance indicates the total number of data fields accepted out of the total number of data fields processed. The Stats procedure expresses acceptance as a percentage. Acceptance is the primary predictor of the savings of automating data entry because accepted fields do not usually require manual verification by an operator. Because the fields do not have a field reject flag set, they are not automatically reviewed in the Remove Flags mode of Completion.

Accuracy indicates the percentage of accepted fields that are correct out of the total number of data fields accepted. The Stats procedure measures accuracy when FormWare generates the recognition results. This occurs pre-Completion, prior to the manual verification of rejected fields. After manual verification, overall accuracy is increased.

Creating Answer Files

When you create a test suite of images, you must also create an answer file for each image file in the test suite. Answer files are ASCII files that contain the correct data produced for each field on a data form. The data must be in the same format that the recognition engine will produce.

Answer files are in a subdirectory named <image path name>.ans that is parallel to your test suite images directory. Although answer files and image files are stored in different subdirectories, each answer file must have exactly the same filename as the corresponding image file in the BDF. For example, if an image file is C:\FormWare\Images\Form0001.tif, the path and name required for the answer file is C:\FormWare\ Images.ans\Form0001.tif.

Note: You should create one answer file and test it before creating multiple answer files.

To create an answer file:

1.   Select a test suite of images and create a test BDF from that set of images.

2. Create a test suite of BDFs for one data form type at a time. The Stats procedure will not report on BDFs with more than one data form type.

3. Determine the output format of the data for all fields.

Note: It is important to set value requirements before testing so that you do not have to edit your answer files later. For example, if a date format should be MMDDYYYY, but you create answer files with dates in MM-DD-YY format, you will not be able to calculate good statistics until you modify the answer files.

4. Using the Designer module, create a job and data form (FDF) to use for testing the data fields. The data fields should have the field names you wish to associate with each answer. Choose the field names carefully so you do not have to edit the answer files later to change the field names.

5. To improve accuracy, use the Designer module to view field properties and verify that each field is defined to allow the input of correct data.

   a. On the Field Properties, Checkbox Edits tab, set the **Verify** field to Key to ensure accurate answer files.

   b. If available, attach VBA or FPL field edits to fields within Designer.

   c. Set date fields to only capture numeric characters and spaces. Separators within date fields should not be captured or allowed.

6. In Administration, open the Rulclnt.ini file in the \FormWare\INI\System folder. This file is added during installation. Uncomment the FW_RUL_ANSWERS 1 parameter by removing the two dashes before it. If this file or parameter do not exist, create a Rulclnt.ini file with only this parameter.

7. From JobFlow Designer, create a new JobFlow and add any needed tasks such as Image Pre-Processing, Enhancement, Form ID, and Recognition. Add the following tasks to the JobFlow.

   a. Add a Completion - New task to the JobFlow to capture data from the images in the test BDF. Information for the answer files is created from the data captured during Completion.

   b. Add a Completion - Verify task to the JobFlow. The quality of the answer files is important for development because they are used to fine tune recognition results. You can revise answer files during recognition development, but we recommend that you include the Completion-Verify task in your JobFlow to ensure that all data is correct before answer files are generated.

   c. Add an Edits task to the JobFlow. On the Edit Types dialog, select the **InScript Rules** check box and select **OK**. Display the Properties dialog for this task and ensure that the Client Parameters field on the General tab is blank.

   d. Save your changes and exit JobFlow Designer.

8. Using the JobFlow Client, run the test suite of images through the JobFlow. While keying a batch, keep a list of recognition problems you encounter and the TIF file on which they occur. You can then tune the rules using the Test Recognition option in Designer (on the Job Properties, Test tab). Keep doing this until your results are satisfactory.

9. When you test your images through the JobFlow, data in those images is captured during Completion. Then, the data file is processed by the Edits task and includes the RULCLNT

parameter set you set. The Edits task generates an answer file for each image with the same name as the image in the BDF, including the .tif extension

An answer file contains field name and field data, with each entry on a separate line, as in the following sample expense form:

```
Name BILL SMITH
Date 04171999
Phone 67.80
Air_total 394.87
Mnday_Total 6411.83
Wed_Dinner 67.41
Airline 197.87
DataField_11 TRADE SHOW EXPENSES
```

Editing Answer Files

The data in an answer file should be the same as the output of Recognition, rather than just the data that appears on the paper form. To ensure accuracy, you may need to edit and correct the answer files generated from FormWare completion data. Use the text editor in Administration to edit your answer files.

1. Open Administration.

2. From the Tools menu, select the **Editor** option to display the text editor.

3. Select the **Open** button on the Editor toolbar.

4. Select for the appropriate answer file and select **Open**. Answer files must contain the correct output for each field on a data form in identical format to the data the recognition engine produces. For example, if the paper form date format is 4/13/02 and the recognition output is to be 04132002, the format for the data in the answer file should be 04132002.

5. The data in each answer file must exactly match the field value, including upper and lowercase characters, unless you specify the /IGNORECASE parameter in the Stats tool argument file. (For information on creating a Stats tool argument file, see the Creating an Argument section that follows.) You can also strip specified characters from both the output and the answer files before they are compared using the /WHITESPACE parameter in the statistics argument file. You must separate field names from the answer, or field data, by one or more blank spaces. For example:

```
fieldnameA answerA
fieldnameB answerB
fieldnameC answerC
```

6. For non-column processed fields, each field name must match the defined field name exactly as it is in the output file, including upper and lowercase characters. Blank non-column processed fields can be omitted from the answer file.

7. For column-processed fields, every instance for which you want output, including blank fields, must have an entry. For every blank instance, you must insert a blank entry in the answer file. If you do not want output, do not include an entry in the answer file.

Running the Stats Procedure

To run the Stats procedure:

1. Generate answer files as described previously using InScript as the recognition engine in the JobFlow.

2. To enable the Stats procedure, set either of the following parameters in the <engine>.ini file (e.g., AEGClnt.ini) as follows:

```
DOSTATS=Y
DOSTATS=<pathname to an arg file>\<formname.sta>
```

3. If you include only the DOSTATS=Y parameter, the Stats procedure will generate the default Stats reports. If you want to customize the reports, specify the path name to a Stats argument file. The <formname.sta> value is the name of the recognition statistics argument file.

4. Place the argument files in the \FormWare\Stats directory.

5. Perform a recognition test from the Job Properties, Test tab in Designer.

Note: When you have included the DOSTATS parameter, the Stats tool runs the completion of recognition for the batch. This process creates the recognition statistics output files in the \FormWare\Tmp\Diagnose\Ins directory. which will contain the three report files: Stats.sum, Stats.err, and Stats.frm.

Creating an Argument File

An argument file is an ASCII text file that contains the arguments to the STATS command. By using optional parameters in the recognition statistics argument file, you can specify:

- A subset of the recognition output file to be processed.
- The comparison of outputs and answers.
- The statistics that are produced.

When adding parameters to an argument file, follow these guidelines:

- Begin each parameter with a forward slash (/) and a keyword. Parameters can appear in any order on the command line or in the argument file.
- Separate argument file parameters using new lines.
- All lines starting with two dashes (--) are treated as comment lines and ignored.

```
-- The following are optional parameters
--
/whitespace=" -/"
/ThresholdFields=Detail_Diagnosis_Code001
/ThresholdFields=Detail_Diagnosis_Code002
/ThresholdFields=Detail_Diagnosis_Code003
/ThresholdFields=Detail_Diagnosis_Code004
/ThresholdFields=Detail_Diagnosis_Code005
/ThresholdFields=Detail_Diagnosis_Code006
/IgnoreCaseFields=PatientName
/CompareByValue=Charges001,Charges002,Charges003
/CompareByValue=Charges004,Charges005,Charges006
/ExcludeFields=PageSetup,PageSetup001,Patient_City
/ExcludeFields=Patient_Street,Patient_Zip
/ExcludeFields=Services_Rendered_Address
/ExcludeFields=Charges007,Charges008,Charges009
```

You can use the following parameters within argument files:

- /WHITESPACE = "<C><C>...<C>" - Strips all characters between the double quotation marks from both the answer and output files before comparing the two fields.
- /FIELDOUTPUT = NONE|BASIC - Prevents or limits the field-by-field summary output. FULL is the default.

- /COMPAREBYVALUE = FIELDLIST - Compares numeric fields by value rather than by string. You might want to compare by value those numeric fields that were entered in the answer files in free format but output in standard formats. For example, "0" and "0.00" is counted as a correct match if you include this parameter.

- FIELDLIST - List of field names, exactly as they appear in the answer files. A comma separates each field name without spaces. For example, /COMPAREBYVALUE = Charges001,Charges002. If you specify multiple /COMPAREBYVALUE arguments, the FIELDLIST values concatenate.

- /EXCLUDEFIELDS = FIELDLIST - Excludes the fields in the FIELDLIST parameter from recognition statistics processing. If you specify multiple /EXCLUDEFIELDS arguments, the FIELDLIST values concatenate.

- /IGNORECASEFIELDS = FIELDLIST - Ignores the case (upper or lowercase) of the listed fields when comparing the output and the answer files. If you specify multiple /EXCLUDEFIELDS arguments, the FIELDLIST values concatenate.

- /THRESHOLDFIELDS = FIELDLISTOPT - This parameter causes the Stats tool to recommend THRESH (threshold) rule settings for the listed fields. If you specify multiple /THRESHOLDFIELDS arguments, the FIELDLISTOPT values concatenate.

- FIELDLISTOPT - An expanded version of FIELDLIST, FIELDLISTOPT allows you to specify a Cost Differential Factor (CDF) immediately following each field name. The CDF calculates the cost of rejecting incorrect fields at the expense of rejecting correct fields. A high CDF means high accuracy. Specify the cost differential factor by immediately following the field name with a colon and the cost differential factor number. For example: thresholdfields = TOTAL:5,PatientName, where TOTAL uses a cost differential factor of 5 and PatientName uses the default cost differential factor of 2. A cost differential factor of 2 specifies that it is twice as important to prevent accepting incorrect field data than to prevent rejecting correct field data. It is acceptable to send up to two correct fields for verification, if by doing so, one incorrect field would also reject for verification instead of being erroneously accepted.

Reviewing Recognition Statistics Reports

The following recognition statistics reports are available within FormWare:

- Form Summary (Stats.sum), which contains the summary statistics for the entire output file and for each field across all selected data forms.

- Error (Stats.err), which contains a listing of all the fields explicitly rejected or erroneously accepted by the recognition engine.

- Field Summary (Stats.frm), which contains the summary statistics for all fields on each data form.

The Stats tool generates all three reports upon execution and displays error messages if it encounters problems while processing. The type of information and the level of detail that appears in each report is determined by parameters that you set in a Stats tool argument file referenced on the DOSTATS= line in each recognition engine's INI file. For example, you can exclude fields or request a recommended threshold for a field.

The following figure shows input and output from the Stats tool comparing the recognition output with the information in the answer files and producing the three reports shown in the diagram.
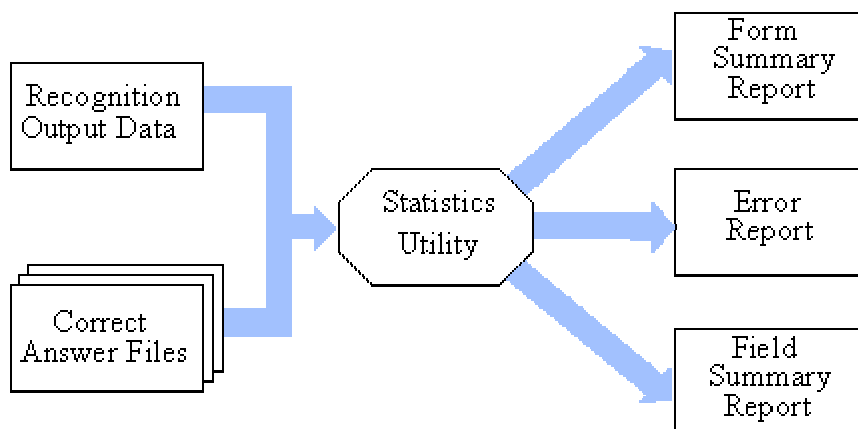
Each report is created in the \FormWare\Tmp\Diagnose\XXX directory, where XXX is the three-letter prefix for the recognition engine that created the output for that field (e.g., INS, AEG, RSP, MIT, PAR, RUL, or CGK). The reports have a base filename of Stats, but with different extensions.

Reviewing Error Reports

You generate Error reports (ERR) with the Stats tool to show a list of every field that:

- Has been rejected by rules.
- Contains output that did not match its corresponding answer.
- Contains no output, but has a corresponding non-blank answer.

The report contains a header for each registered image processed and a list of the fields on that form that contain errors. Each field error appears on a single line. The following example is a block for a page that contains one field error:

```
IMAGENAME: C:\FORMWARE\IMAGES\FORM0006.004
DOS003: failed=1, correct=0, answer=013092 output=01-13-09 errormsg=(3)
Subst rule failed; nothing matched.
```

Consolidate error reports using the FINDERR utility. You can see the following fields on an Error report:

- **<imagename>** - Name of the image file (including path and filename) that contains the field with the error.
- **<fieldname>** - Name of the field that was in error.
- **Failed** - Indicates whether the field was rejected (failed=1) or accepted (failed=0).
- **Correct** - Indicates whether the field matched its answer (correct=1) or did not (correct=0).
- **Answer** - Answer in the answer file, or implied blank if no answer displays.
- **Output** - Field data produced by the recognition engine, or implied blank if no output appears.
- **Errmsg** - Description of the field error as it appears in the recognition output file.

Using the FindErr Utility

The FindErr utility resides in the \FormWare\Programs directory. It can report all errors for one field across all of the form images in a batch and also reports the filenames of all images that failed for the specified field. This helps to detect patterns of errors for a single field.

To use the FindErr utility:

1. Run the Stats tool and generate an error report.

2. Run the FindErr utility by typing this command and parameter at a DOS command prompt from the FormWare\Tmp\Diagnose\Ins folder:

    ```
    _FINDERR.EXE TAG=field name stats.err > finderr.out
    ```

3. For example:

    ```
    _finder=DOS001 stats.err > finderr.out
    ```

4. In this example, the field name (DOS001) indicates the field for which you want a report. The field name is case-sensitive and must exactly match the case of the defined field name. For column-processed fields, the field name in the answer file must have the instance number of the field appended as a three digit, zero left-padded number. Stats.err is the name of the Recognition Statistics Error report that will be used by the FindErr utility. FindErr.out is the name of the file to which the FindErr utility writes the field errors.

Reviewing the Form Summary Report

Depending on the parameters in the Stats argument file and the value in the DOSTATS= parameter in the recognition engine INI file, the Form Summary report (SUM file) can contain the following information:

*   Summaries of the total images processed, including acceptance and accuracy rates.
*   Basic information on the total number of accepted and accurate fields.
*   Detailed information on blank, non-blank, and rejected fields.
*   Recommended THRESH (threshold) rule settings for fields.

The Form Summary report begins with a header describing the recognition statistics command parameters. Following is an example of the first section of a Form Summary report:

```
Recognition Form Statistics
-------------------------------------
09/24/02  16:27:30
-------------------------------------
Input Parameters
-------------------------------------
```

Form:	HCFARED
Page:	
Output Batch:	
Answers:	c:\FORMWARE\IMAGES.ans
FDF:	
Whitespace:	' -/'
FieldOutput:	Full
CompareByValue Fields:	Charges001,Charges002,Charges003,Charges004,Charges005,Charges006,TOTAL,PAID
Exclude Fields:	PRIOR_AUTHORIZATION,Charges007,Charges008,Charges009,Charges010,Charges011,Charges012
IgnoreCase Fields:	PatientName
Threshold Fields:	ACCOUNT_NUMBER,Charges001,DOS001,Days_Units001,Detail_Diagnosis_Code001,DOS002,DOS003,DOS004,DOS005,DOS006

The second section of the Form Summary report contains information that summarizes all processed images and all fields on the processed forms. It includes these fields:

- **Count** - Occurrences of each category of data.
- **%** - The Count expressed as a percentage of another count, described in the note column.
- **Note** - Footnote reference that describes the exact meaning of the Count and % data.

Following is an example of the statistics (second) section of a Form Summary Report:

	Count		%		Note
Fields Output					
Total Fields:	27013		100.00%		(a)
Accepted:	23557		87.21%		(b)
Accepted and Correct:	21555		91.50%		(c)
Forms					
Total Pages:	994		100.00%		
Registered:	900		90.54%		
Blank Field Analysis					
Accepted Nonblank:	20195		85.73%		(g)
Accepted Blank:	3362		14.27%		(h)
Accepted and Correct Nonblank:	18698		86.75%		(i)
Accepted and Correct Blank:	2857		13.25%		(j)
Rejected Field Analysis					
Total Fields:	3456		12.79%		(k)
Rejected but Correct:	1016		29.40%		(l)
Rejected Nonblank:	3111		90.02%		(m)
Rejected Blank:	345		9.98%		(n)
Rejected but Correct Nonblank:	808		79.53%		(o)
Rejected but Correct Blank:	208		20.47%		(p)
Column Processing Fields Not Output					(d)
Nonblank Answers:	133		N/A		(e)
Blank Answers:	0		N/A		(f)

The following table describes the codes in the Note column of the Form Summary report:

Note	Count Column	Percent Column
(a)	Total fields produced.	Not applicable.
(b)	Total fields accepted.	Percent of total fields accepted of the total fields produced
(c)	Accepted fields that are correct.	Percent of accepted and correct fields of the total accepted fields produced.

Note	Count Column	Percent Column
(d)	Column-processed fields that created fewer outputs than there are answers.	Not applicable.
(e)	Nonblank answers that do not have any field output.	Not applicable.
(f)	Blank answers that do not have any field outputs.	Not applicable.
(g)	Accepted fields that are not blank.	Percent of fields that are not blank of the total accepted fields.
(h)	Accepted fields that are blank.	Percent of fields that are blank of the total accepted fields.
(i)	Total accepted and correct fields that are not blank.	Percent of fields that are not blank of the total accepted and correct fields.
(j)	Total accepted and correct fields that are blank.	Percent of blank fields of the total number of accepted and correct fields.
(k)	Total rejected fields.	Percent of fields rejected of the total fields.
(l)	Rejected fields that are correct.	Percent of fields that are correct of the total fields.
(m)	Total rejected fields that are not blank.	Percent of fields that are not blank of the total number of rejected fields.
(n)	Total rejected fields that are blank.	Percent of rejected fields that are blank of the total rejected fields.
(o)	Total rejected but correct fields that are not blank.	Percent of rejected but correct fields that are not blank of the total rejected but correct fields.
(p)	Total rejected but correct fields that are blank.	Percent of the total rejected but correct fields that are not blank of the total rejected but correct fields.

Reviewing Field Summary Statistics

Field summary statistics are produced when you start the Stats tool and include the /FIELDOUTPUT=BASIC or /FIELDOUTPUT=FULL parameters in the argument file, although the FIELDOUTPUT parameter is not required. After a header, one line of output appears for each field.

Field	Total (q)	Accept Count (r)	% (s)	Accept and Accept Count (t)	Accept and Correct % (u)	Col Process FldsNot Output Correct Nonblank (v)	Blank (w)
Charges001	895	803	89.72	790	98.38	5	0
Charges002	642	553	86.14	538	97.29	0	0
Charges003	307	245	79.80	223	91.02	0	0
Charges004	188	136	72.34	127	93.38	1	0
Charges005	113	83	73.45	75	90.36	0	0
Charges006	73	40	54.79	34	85.00	5	0
DOS001	895	781	87.26	767	98.21	5	0
DOS002	642	545	84.89	537	98.53	1	0
DOS003	307	241	78.50	231	95.85	0	0
DOS004	188	136	72.34	126	92.65	1	0

Field	Total (q)	Accept Count (r)	% (s)	Accept and Accept Count (t)	Accept and Correct % (u)	Col Process FldsNot Output Correct Nonblank (v)	Blank (w)
DOS005	113	77	68.14	71	92.21	0	0
DOS006	73	40	54.79	33	82.50	5	0

The last line of the header contains a footnote reference that describes the exact meaning of each column. Following are the codes in the Note column:

Note	Footnote Description
(q)	Occurrences of the field output.
(r)	Accepted occurrences.
(s)	Percent of accepted occurrences of the total occurrences.
(t)	Accepted and correct occurrences.
(u)	Percent of accepted and correct occurrences of accepted occurrences.
(v)	Applicable to column processing fields only. Occurrences of the field for which there are non-blank answers, but no output.
(w)	Applicable to column processing fields only. Occurrences of the field for which there are blank answers, but no output.

Reviewing Field by Field Statistics

Field by field statistics are produced when you start the Stats tool and include the /FIELDOUTPUT=FULL parameter in the argument file, although this parameter is not required.

The following is a set of detailed statistics for each field.

```
------------------------------
 Field by Field Statistics
------------------------------
```

Charges001	Count	%	Correct Count	Correct %	
Accepted	803	89.72	790	98.38	(x)
Nonblank	803	100.00	790	100.00	(y)
Blank	0	0.00	0	0.00	(z)
Rejected	92	10.28	23	25.00	(aa)
Nonblank	78	84.78	22	95.65	(bb)
Blank	14	15.22	1	4.35	(cc)

At the end of each line is a footnote reference that describes the exact meaning of each line. In the following notes, the word "occurrences" refers to single fields and instances of column-processed fields.

Note	Count Column	Percent Column	Correct Count Column	Correct Percent Column
(x)	Accepted occurrences.	Percent of accepted occurrences of total occurrences.	Accepted and correct occurrences.	Percent of correct occurrences of accepted occurrences.

Note	Count Column	Percent Column	Correct Count Column	Correct Percent Column
(y)	Nonblank accepted occurrences.	Percent of nonblank occurrences of total accepted occurrences.	Nonblank occurrences.	Percent of nonblank occurrences of total accepted and correct occurrences.
(z)	Blank accepted occurrences.	Percent of blank occurrences of total accepted occurrences.	Blank occurrences.	Percent of blank occurrences of total accepted and correct occurrences.
(aa)	Rejected occurrences.	Percent of rejected occurrences of total occurrences.	Rejected but correct occurrences.	Percent of correct occurrences of rejected occurrences.
(bb)	Nonblank rejected occurrences.	Percent of nonblank occurrences of total rejected occurrences.	Nonblank occurrences.	Percent of nonblank occurrences of rejected and correct occurrences.
(cc)	Blank rejected occurrences.	Percent of blank occurrences of total rejected occurrences.	Blank occurrences.	Percentage of blank occurrences of rejected and correct occurrences.

Using the THRESH Rule

The THRESH rule is occasionally used as a part of a field validation rule to specify required character recognition confidence and spread values. The Stats tool can analyze recognition output and compute optimal threshold settings to adjust the trade off between field acceptance and field accuracy. The Stats tool recommends a THRESH rule setting for a field when you include the /THRESHOLDFIELDS parameter in the argument file.

You can also specify a cost differential factor when you include the /THRESHOLDFIELDS parameter in the argument file. A cost differential factor indicates the cost of rejecting incorrect fields at the expense of rejecting correct fields or vice versa. The higher the cost differential factor, the higher the accuracy, and the greater number of rejected fields there will be.

The Stats tool generates three lines of output for each field that immediately follows the basic or detailed field output. If you set the cost differential factor to 2, the following output is possible (Note that "19/10" is within a cost factor of 2.):

```
Recommended THRESH Rule Settings: confidence=100, spread=0, count=16%
10 Accepted and Incorrect would be rejected,
19 Accepted and Correct would be rejected
```

To make use of the recommended settings, edit the rule file and add the THRESH rule as the last line before accepting the field. If you ACCEPT the field at multiple points in the field's rule, THRESH should be added at each point.

The three recommended settings for the THRESH rule for this field are:

- confidence=#
- spread=#
- chars=%#

"# Accepted and Incorrect" is the number of currently accepted but incorrect fields that the Stats tool will reject if the recommended THRESH rule settings are used. "# Accepted and Correct" is the number of accepted and correct fields that the Stats tool will reject if the recommended

THRESH rule settings are used. This is the cost of rejecting some correct fields to reject incorrect fields.

If the field is expressly excluded from output when you include either the /FIELDOUTPUT=NONE or /EXCLUDEFIELDS=*FIELDNAME* parameter in the argument file, a threshold setting is not produced, even if include the /THRESHOLDFIELDS argument.

Reviewing the Field Summary Report

The Field Summary report (FRM file) shows the accuracy of each registered image. This report contains one line for each image. You can use this report to find images that have poor overall accuracy. Such images may indicate a badly tuned scanner, a poorly registered image, or a form variant not handled by the FDF.

The Stats tool sorts the report from lowest to highest accuracy, where the accuracy is the number of correct fields divided by the total number of fields on the data form.

The following example is for an output file of three registered forms.

```
C:\FORMWARE\IMAGES\FORM0006.3E2 34 correct, 5 wrong -- 87.18
C:\FORMWARE\IMAGES\FORM0006.3E1 47 correct, 6 wrong -- 88.68
C:\FORMWARE\IMAGES\FORM0006.3E3 55 correct, 5 wrong -- 91.67
```

This example contains the following values:

- **Filespec** - Path of the image file from which the form was recognized.
- **# correct** - Number of fields on the image with correct output.
- **# wrong** - Number of fields on the image with incorrect output (or those fields that should have been produced but were not).
- **- #** - Percentage of accurate fields of the total number of fields processed.

Reviewing the Maximum Output File

The Stats tool generates this file as a diagnostic aid. It is located in the C:\FormWare\Tmp\Diagnostics\XXX directory (where "XXX" is a recognition engine prefix) when you set the environment variable FW_XXX_INSDATA to non-zero.

```
FORM =
{
        FORMNAME = hrgoldma
        FDFNAME = hrgoldma.fdf
        PAGE =
        {
                PAGENO = 0
                DPI = 300
                IMAGENAME = RED2.TIF
                REGISTRATION =
                {
                        DX = -21
                        DY = -10
                        CX = 0
                        CY = 0
                        THETA = 0.007089
                        SCALE = 1.000000
                }
        FIELD =
                {
                        NAME = PageSetup
                        INSTANCE = 1
                        CLIP =
```

```
                                        {
                                                X = 0
                                                Y = 0
                                                WIDTH = 16
                                                HEIGHT = 32
                                        }
                                }
                                FIELD =
                                {
                                        NAME = DOS
                                        INSTANCE = 1
                                        CLIP =
                                        {
                                                X = 80
                                                Y = 110
                                                WIDTH = 485
                                                HEIGHT = 42
                                        }
                                        LINE =
                                        {
                                                BEST =  1   2  -   1   3  -   9   1   ^   1   2
-   1   3   -   9 1
-- 12-13-91 12-13-91
                                                CONF         100 100 100 100 100 100 100 100 100
100 100 100 100 100 100 100
                                                SECOND =          *   *   *   *   *   *   *   *   ^   *   *
*   *   *   *   *   *
-- ******** ********
                                                SPREAD =100 100 100 100 100 100 100 100 100 100
100 100 100 100 100 100
                                                XPOS =        80  109    0 167 196    0 254 283 341
370    0 428 457    0 515 544
                                                YPOS =       110 110    0 110 110    0 110 110 110
110    0 110 110    0 110 110
                                                WIDTH =              21   21    0  21  21    0  21
21  21  21    0  21  21    0  21  21
                                                HEIGHT =      42   42    0  42  42    0  42  42  42
42    0  42  42    0  42  42
                                        }
                                }
                                FIELD =
                                {
                                        NAME = POS
                                        INSTANCE = 1
                                        CLIP =
                                        {
                                                X = 13
                                                Y = 121
                                                WIDTH = 164
                                                HEIGHT = 32
                                        }

        LINE =
                                        {
                                                BEST = 8
-- 181
                                                CONF = 25
                                                SECOND = *
-- ***
                                                SPREAD = 7
                                                XPOS = 72
                                                YPOS = 121
                                                WIDTH = 21
                                                HEIGHT = 32
```

```
                                }
                        }
                        FIELD =
                        {
                                NAME = PROC_CODE
                                INSTANCE = 1
                                CLIP =
                                {
                                        X = 118
                                        Y = 110
                                        WIDTH = 214
                                        HEIGHT = 63
                                }
                                ERROR = (3) code must be 9,7, or 5 chars long

                LINE =
                                {
                                        BEST =    @  @  )
        -- @@)
                                        CONF =   25   25   25
                                        SECOND =    *   *   *
        -- ***
                                        SPREAD =    7    7    7
                                        XPOS =   118 206 312
                                        YPOS =   110 118 125
                                        WIDTH =   93 117   20
                                        HEIGHT =  55   55   40
                                }
                        }
                FIELD =
                        {
                                NAME = CHARGES
                                INSTANCE = 1
                                CLIP =
                                {
                                        X = 166
                                        Y = 110
                                        WIDTH = 141
                                        HEIGHT = 42
                                }

                LINE =
                                {
                                        BEST =    6  5  .  0  0
        -- 65.00
                                        CONF =   100 100 100 100 100
                                        SECOND =    *   *   *   *   *
        -- *****
                                        SPREAD =  100 100 100 100 100
                                        XPOS =    166 206   0   0   0
                                        YPOS =    110 110   0   0   0
                                        WIDTH =    21  21   0   0   0
                                        HEIGHT =   42  42   0   0   0
                                }
                        }
                        FIELD =
                        {
                                NAME = SEX
                                INSTANCE = 0
                                CLIP =
                                {
                                        X = 0
                                        Y = 0
                                        WIDTH = 0
```

```
                                        HEIGHT = 0
                            }

            LINE =
                            {
                                    BEST =    F  e  m  a  l  e
        -- Female
                                    CONF =    100 100 100 100 100 100
                                    SECOND =   *   *   *   *   *   *
        -- ******
                                    SPREAD =  100 100 100 100 100 100
                                    XPOS =      0   0   0   0   0   0
                                    YPOS =      0   0   0   0   0   0
                                    WIDTH =     0   0   0   0   0   0
                                    HEIGHT =    0   0   0   0   0   0
                            }
                    }
                    FIELD =
                    {
                            NAME = NAME
                            INSTANCE = 0
                            CLIP =
                            {
                                    X = 0
                                    Y = 0
                                    WIDTH = 880
                                    HEIGHT = 128
                            }
                            ERROR = (3) Thresh rule failed; too many chars with low
        conf/spread (1).
            LINE =
                            {
                                    BEST =    S  E  T  T  E  L  E  ,  ^  M  A  R  Y
        -- SETTELE, MARY
                                    CONF =    100 100 100 100 100 100 100  25    100
        100 100 100
                                    SECOND =   *   *   *   *   *   *   *   *   ^   *   *   *   *
        -- ******** ****
                                    SPREAD =  100 100 100 100 100 100 100   7    100
        100 100 100
                                    XPOS =     68   97 126 155 184 213 242 279   329
        358 387 416
                                    YPOS =     49   49  49  49  49  49  49  81    49
        49  49  49
                                    WIDTH =    17   17  17  17  17  17  17   8    17
        17  17  17
                                    HEIGHT =   38   38  38  38  38  38  38  12    38
        38  38  38
                            }
                    }
            FORM =
            {
                    ...
            }
```

Generating Production Statistics

The purpose of production statistics is to collect and log recognition engine statistics at the engine, job, form, and field level for an individual recognition engine. Statistics can only be gathered for a single instance of a recognition engine. If you have multiple engines performing recognition, each would generate its own statistics file.

You can generate cumulative statistics on a daily basis at specified time or between engine start up and shutdown. If the engine is shutdown before the specified time, then engine shutdown stats will be reported instead. You use INI settings to specify the frequency with which you want recognition statistics to be generated.

> **Note**: Daily statistics require that the recognition engine be run in transaction mode. The OCX version of the engine must be running from JobFlow and remain active during the specified time.

Session statistics output cumulative information on all jobs, forms, and fields for the period of execution of the recognition engine. If you run the engine from JobFlow using the EXE version, then a statistics file is generated for each BDF. If you execute the engine using the OCX version, then the statistics file covers all BDFs processed during the period that the engine was running, from startup until shutdown.

Enabling Statistics

You generate production statistics and their frequency through an INI setting. Statistics files can grow large quickly; therefore, the default behavior of FormWare is to not generate these files. You add the INI parameters used to enable and specify the frequency of recognition statistics in the System.ini file located in the \FormWare\INI\System directory, or you can put them in the INI file for the recognition engine (e.g., InsClnt.ini for InScript).

The INI parameters relating to production statistics are described in the following table.

Parameter	Description
FWRECOGSTATS=<DLLName>	Enables statistics generation with the specified DLL name. To use the FormWare DLL, the INI file setting should be: FWRECOGSTATS=FWRESTAT.DLL
FWRESTATFREQ=<Frequency>	Determines the frequency with which recognition statistics should be generated using the DLL in the FWRECOGSTATS parameter. <Frequency> can be one of the following values: DAILY/HHMM ONSHUTDOWN FWRESTATFREQ=ONSHUTDOWN FWRESTATFREQ=DAILY/1200 (daily at noon) If you have enabled statistics, but you have not specified a frequency in the INI file, the default frequency is ONSHUTDOWN.
FWRESTATFILENAMES=<Value>	Determines the naming convention of the statistics files. <Value> can be LONG or SHORT. Refer to the Filenames and Location section that follows for information on statistics filenames.

Understanding File Names and Locations

Each generated statistics file receives a unique name, which is derived from the machine name, Julian date, and time, regardless of the frequency chosen. The files are in the \FormWare\Statistics directory. You can view them from Administration using a text editor.

The structure for a statistics file name is:

```
FWRESTAT_<machine name>_JJJ_HHMMSS.txt
```

where JJJ is the Julian date and HHMMSS is the time in hours, minutes, and seconds when the file is created.

The default file naming convention for statistics files exceeds 8 characters and, therefore, can present problems for users on certain versions of Novell operating systems. For that reason, you can specify whether long or short names should be used for statistics file names. If you set FWRECOGSTATFILENAMES=SHORT in your System.ini or recognition engine INI file, portions of the name are converted into directories, rather than being used as part of the filename. For example, a file generated on a PC named RECOSRV could be named as follows:

```
FWRESTAT\RECOSRV\141_1230.txt
```

Understanding Recognition Statistic File Format

The recognition statistics are organized in a hierarchy of data objects. Each object at a given level contains a linked list of objects at the level below. The engine is the highest level in the hierarchy, followed by job, form, and field. The engine can link to one or more jobs, each job can link to one or more forms, and each form can link to one or more fields. This hierarchy determines the organization of the data in the recognition statistics files.

The following table lists the statistical items maintained for the engine, job, and form. Items are tallied for each of the three levels unless otherwise indicated.

Item	Note
Number of Batches	Tallied at engine and job level only.
Number of Jobs	Tallied at engine level only.
Number of Forms	Tallied at job level only.
Number of Fields	Tallied at form level only.
Seconds Recognizing	N/A
Number of Images	N/A
Number of Processed Fields	Fields processed by the current engine. Fields processed by a different recognition engine or KFI fields are not included in this number.
Number of HP fields	Handprint fields processed by this engine.
Number of MP fields	Machine print fields processed by this engine.
Number of MS fields	Mark sense (OMR) fields processed by this engine.
Number of Accepted fields	Un-flagged fields.
Percentage of Fields Accepted	(# Accepted) ÷ (# Processed) Fields can be accepted by applying rules or VBA edits; therefore, it is possible for this value to exceed 100.
Number of Chars Processed	Number of characters that actually are written to the recognition results file.
Number of Chars Recognized	Characters above confidence threshold as specified on the Field Properties, Recognition tab in Designer.
Number of Chars Rejected	Characters below confidence threshold as specified on the Field Properties, Recognition tab in Designer.
Percentage of Chars Recognized	N/A
Number of HP Chars	Handprint characters processed by this engine.
Number of Rejected HP Chars	Handprint characters that fall below the confidence threshold.

Item	Note
Percentage of HP Chars Recognized	N/A
Number of MP Chars	Machine print characters processed by this engine.
Number of Rejected MP Chars	Machine print characters that fall below the confidence threshold.
Percentage of MP Chars Recognized	N/A
Number of MS Chars	Mark sense characters processed by this engine.
Number of Rejected MS Chars	Mark sense characters that fall below the confidence threshold.
Percentage of MS Chars Recognized	N/A

The following table lists the statistical items maintained at the field level.

Item	Note
Field Name	Field name as it appears on the Field Properties, General tab in Designer.
Field Type	Recognition type for this field. Possible values are HP (Handprint), MP (Machine print), and MS (Mark sense).
Num Times Processed	Depends on the number of images of the form type to which this field belongs.
Num Chars Processed	Number of characters that actually are written to the recognition results file.
Num Chars Recognized	Characters above confidence threshold as specified on the Field Properties, Recognition tab in Designer.
Num Chars Rejected	Characters below confidence threshold as specified on the Field Properties, Recognition tab in Designer.
Num Times Field Accepted	N/A
Percent Chars Rejected	(# Chars Rejected) ÷ (# Chars Proc'd)
Percent Fields Rejected	(# Times Proc'd - # Times Field Accepted) ÷ (# Times Proc'd)

Following is a sample Statistics file.

```
***********************************************************
*            FormWare Engine Statistics                   *
***********************************************************


Date/Time:            Wed Feb 17 13:50:17 1999

Number of Batches =                       4
Number of Jobs =                          2
Seconds Recognizing =              4.563000
Number of Images =                       14
Number of Processed Fields =             28
Number of HP Fields =                    14
Number of MP Fields =                    10
Number of MS Fields =                     4
Number of Accepted Fields =              28
Percentage of Fields Accepted =      100.00

Number of Chars Processed =             198
```

```
Number of Chars Recognized =              198
Number of Chars Rejected =                  0
Percentage of Chars Recognized =       100.00

Number of HP Chars =                       84
Number of Rejected HP Chars =               0
Percentage of HP Chars Recognized =    100.00

Number of MP Chars =                      110
Number of Rejected MP Chars =               0
Percentage of MP Chars Recognized =    100.00

Number of MS Chars =                        4
Number of Rejected MS Chars =               0
Percentage of MS Chars Recognized =    100.00

============ Start of Job Statistics for Job (STATS) =================

Number of Batches =                         2
Number of Forms =                           2
Seconds Recognizing =                    3.53
Number of Images =                         10
Number of Processed Fields =               20
Number of HP Fields =                      10
Number of MP Fields =                       6
Number of MS Fields =                       4
Number of Accepted Fields =                20
Percentage of Fields Accepted =        100.00

Number of Chars Processed =               142
Number of Chars Recognized =              142
Number of Chars Rejected =                  0
Percentage of Chars Recognized =       100.00

Number of HP Chars =                       60
Number of Rejected HP Chars =               0
Percentage of HP Chars Recognized =    100.00

Number of MP Chars =                       78
Number of Rejected MP Chars =               0
Percentage of MP Chars Recognized =    100.00

Number of MS Chars =                        4
Number of Rejected MS Chars =               0
Percentage of MS Chars Recognized =    100.00

+++++++++++++ Start of Form Statistics for Form (STATS002)
+++++++++++++++++

Number of Fields =                          6
Seconds Recognizing =                    1.91
Number of Images =                          4
Number of Processed Fields =                6
Number of HP Fields =                       3
Number of MP Fields =                       0
Number of MS Fields =                       3
Number of Accepted Fields =                 8
Percentage of Fields Accepted =        133.33

Number of Chars Processed =                28
Number of Chars Recognized =               28
Number of Chars Rejected =                  0
Percentage of Chars Recognized =       100.00
```

```
         Number of HP Chars =                         24
         Number of Rejected HP Chars =                 0
         Percentage of HP Chars Recognized =      100.00

         Number of MP Chars =                          0
         Number of Rejected MP Chars =                 0
         Percentage of MP Chars Recognized =        0.00

         Number of MS Chars =                          4
         Number of Rejected MS Chars =                 0
         Percentage of MS Chars Recognized =      100.00


         ------------------ Start of Field Statistics ------------------
                              Num        Num        Num        Num        Num
         Times    Percent    Percent
                              Times      Chars      Chars      Chars      Field
         Chars     Fields
           Field Name        Type   Proc'd     Proc'd     Recog'd    Rejected
         Accepted   Rejected  Rejected
         ---------------      ----   ------     ------     -------    ---------  ------
         --    --------  --------
         STATS002_Fld0002     HP     4          24         24         0          4
         0.00      0.00
         STATS002_Fld0001     MS     4          4          4          0          4
         0.00      0.00
```

Setting Up QA Statistics

Once you install QA Statistics on the FormWare File System, you can set parameters that determine which jobs, forms, and fields are enabled to collect this type of statistical data and how long this information should be kept in the database (i.e., statistic purge parameters).

Before you can collect statistics at the job, form, or field level, you must enable it to do so. By default, QA statistics collection is disabled for all existing jobs. The steps in this task show you how to access an existing job and enable it to capture QA statistics.

The steps in this task assume that you have existing jobs defined for FormWare and have installed the QA Statistics module as part of a File System Installation. See Installing QA Statistics in the Installing & Configuring section.

To access the QA Statistics module and set parameters:

1. From the desktop, select **Start > Programs > FormWare > QA Statistics Console**.

2. Select the QA Statistics icon in the Administration drawer to open the QA Statistics window.

    The QA Statistics window is divided into three sections (from left to right): program bar, tree view, and display area. Select icons and drawers from the program bar to access QA Statistics applications. Within the tree view, select the plus (+) sign or the minus (–) sign next to a node, to expand or collapse it. The top level in the tree view designates a job, forms within that job, and then fields within that form. When you select a node in the tree view, detail for that item displays in the display area.
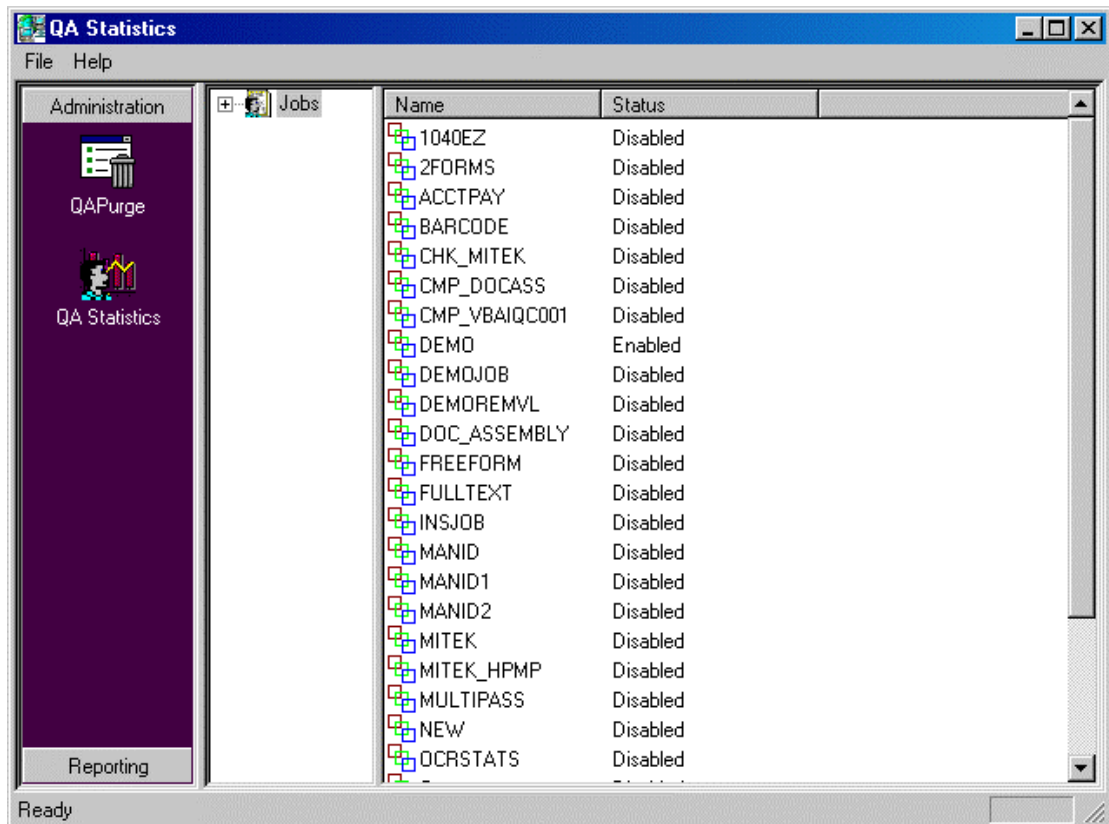
3. Expand the existing jobs in the tree view by clicking on the + sign next to the Jobs node. From here you have the following choices:

   o To enable an entire job, highlight one or more job names in the display area (not the tree view), right-click to display the contents menu, and select **Enable**.
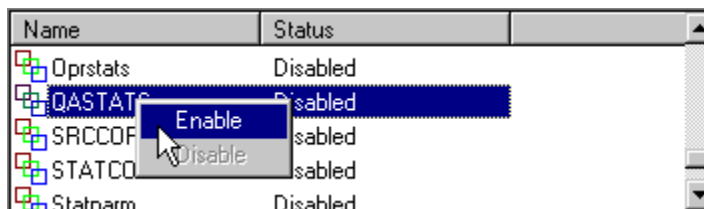
   o To enable forms within a job, select the job name in the tree view to expand it, highlight one or more form names in the display area, right-click to display the contents menu, and select **Enable**.
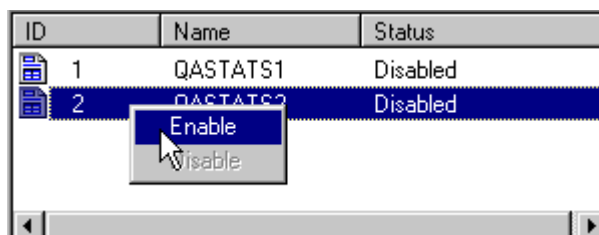
   o To enable a field within a form, select the form name in the tree view to expand it, highlight one or more field names in the display area, right-click to display the contents menu, and select **Enable**.

ID		Name	Status
[ABC]	1	QAS_Fld_01	Disabled
[ABC]	2	QAS_Fld_02	Disabled
[ABC]	3	QAS_Fld_03	Disabled
[ABC]	4	QAS_Fld_04	Disabled
[ABC]	5	QAS_Fld_05	Disabled
[ABC]	6	Fld_06	Disabled
[ABC]	7	Fld_07	Disabled
[ABC]	8	QAS_Fld_08	Disabled
[ABC]	9	QAS_Fld_09	Disabled
[ABC]	10	QAS_Fld_10	Disabled
[ABC]	11	QAS_Fld_11	Disabled

Enable
Disable

Note: Enabled jobs, forms, and fields can be disabled using this same procedure.

4.   Select **File > Exit** to close the QA Statistics module.

Defining a JobFlow to Collect QA Statistics

Once you have enabled individual jobs, forms, and fields to collect QA statistics, you must create a JobFlow with components that are capable of capturing this data.

The QA Statistics module requires that all FormWare components for which you want to capture statistical data be run in OCX mode. When a FormWare component is run in OCX mode, it remains open and waiting to process work, as opposed to closing each time a batch has completed processing.

Follow these steps to create a JobFlow for QA statistics collection:

1.   From Administration, launch JobFlow Designer. The JobFlow Designer window displays.

2.   Select **New** on the toolbar to display the New JobFlow dialog.

3.   In the **New JobFlow Name** field, type the name of this JobFlow and then **OK**. The JobFlow name displays in the JobFlow Browser.

4.   Define the tasks in the JobFlow in the order that you want them to process.

5.   For each task that you want to capture QA statistics, display the Task Properties dialog for the task.

   a.   In the **Client Program Name** field, change the existing program name to the OCX version of that program. For example, change "Complete" to "Complete.ocx".

   b.   In the **Client Parameters** field, add the "-FWSTATS=Y" switch after any existing parameters for this component.

6.   Here is the Task Properties dialog for a Completion – New task:
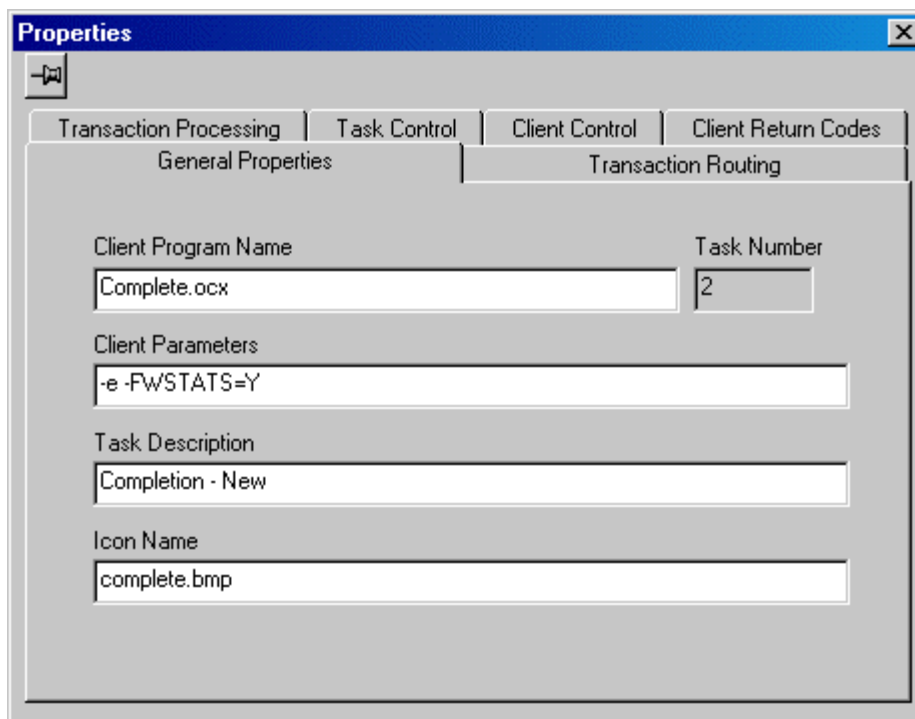
Note: You only need to set the OCX mode on those tasks where you want to capture QA statistical data. Do not set these parameters for FormWare components (such as Form ID) where the collection of this data is meaningless and will produce unnecessary processing overhead.

7. Save your changes and then **File > Exit** to exit JobFlow Designer.

Proceed to Generating QA Statistics Reports in the Maintaining section for instructions on how to generate reports on this statistical data.

Writing Code to Apply Business Rules

The \FormWare\Rules directory is the repository for rules files, which you can identify by their RUL extension. Rules files are used in validation and interrogation of both recognition results and batch data (i.e., pre-Completion and post-Completion). The InScript FDF is also saved in the **Rules** folder. InScript FDF is a text file that defines the recognition zones and parameters for a given form created in Designer.

A rules file is an ASCII file containing data acceptance procedures written in a fourth-generation rule language created by FormWare. Rules files work with FDFs, which contain the specifications needed to process batches of a specific form type. The FDFs listed in the Rules folder control all form processing and recognition and reference the rules file for acceptance requirements. A single FDF can reference no more than one rules file, but several FDFs can all reference the same rules file.

Modifying Job Properties for Completion

Creating Job-Level Help

A job help file is displayed during Completion whenever the operator presses the [F5-Esc] keys. This help is a text file that is stored in the \FormWare\Jobs folder. The name of the file is the same as the job name with an .hlp extension.

To create a job help file:

1. Open the job in Designer.

2. Open the Job Properties dialog and select the General tab.

3. Select the **Help Text** button. This launches the FormWare text editor, which displays the name and path of the help file in the title bar.

4. Type the help text you want displayed to the operator for the job.

5. Select the **Save** button within the text editor to save your changes.

6. Select **File > Exit** to close the text editor and return to Designer.

Editing the Job INI File

Each job can have its own INI file that defines special parameters used during Completion or other processes by that job. The job INI files are located in the \FormWare\INI\Jobs folder and have the name of the job with a .ini extension. You can view job INI files within Designer or from Administration.

To add or modify the INI file for the active job within Designer:

1. Open an existing job or create a new one in Designer.

2. Display the Job Properties, General tab for the job.

3. Select the **Edit .INI** button. The job INI file displays in the text editor.

4. Type new parameters or modify the existing parameters in the INI file. For a list of the available INI parameters, see INI Parameters in the Supplementary Information section.

5. When you are finished, select the **Save** button on the text editor toolbar to save your changes.

6. Select **Exit** from the File menu to return to Designer.

7. Select **Save All** on the Designer toolbar to save your changes.

Note: See the Setting INI Parameters section for more information on setting INI parameters within FormWare.

Setting Edit Properties for the Job

The Job Properties, Edits page specifies the edit programs used by the current job. The settings you make here can only be viewed within the Completion module. No other components are affected by the parameters on this page.

To set edit properties:

1. In the **File name protection** field, type the prefix with which all data files created by Completion will start. This option does not apply when you use JobFlow to access Completion.

2. In the **Numeric exceptions** field, type the exception characters for numeric fields. These are exception characters that you can type in numeric fields.

3. Select the **Perform type checking** check box to enable field type checking for reject repair. Recognition data is compared against the field type. Any character that does not match is converted to a character reject. Recognition results can return a value not of the field type set for a field, but the data is placed in the field without error. Set this parameter to force a check of the data. For example, if an unsigned numeric field receives results of "209F44568" and you did not select this check box, the data is accepted. If **Perform Type Checking** is selected, the data result is "209?44568."

4. Select the **Maintain record position** check box to prevent record synchronization. This means that the cursor stays on the record moved to by the FPL field edit program.

5. Select the **Run edits on interrupt** check box to cause the FPL File End Edit program to be executed against the batch even when the batch is incomplete.

6. Select the **Always run edits** check box to execute FPL edits whenever a field is exited. By default, the field edits are not executed if the operator selects the down arrow.

7. Select the **Disable VBA project** check box to disable the VBA project associated with the current job (i.e., VBA processing is not executed). This is helpful when testing or troubleshooting your application.

8. Select the **Fire VBA events on enabled fields only** check box to indicate to Completion that VBA processing should occur only for those fields that have been selected for field edits. The default is for Completion to fire VBA events on all fields.

9. In the **Field end edit** field, select the **Browse** button to select from a list of all existing FPL Field Edit programs in the \FormWare\Field Edit folder. Select the **Edit** button modify the syntax of the selected program in a text editor. Select the **Compile** button to parse the program file to convert the syntax of the source code to a binary file, which is then used in processing. This binary file has the same name as the corresponding program file, except with an .r extension.

10. In the **File end edit** field, select **Browse** button to select from a list of FPL File End Edit programs in the \FormWare\File Edit folder. Select the **Edit** button modify the syntax of the selected program in a text editor. Select the **Compile** button to parse the program file to convert the syntax of the source code to a binary file, which is then used in processing. This binary file has the same name as the corresponding program file, except with an .r extension.

11. Select **Save All** on the Designer window toolbar to save your settings.

Setting Option Properties for the Job

Using Designer, you can set automatic options that affect Completion on the Job Properties, Options tab. These options include the auto and oversign functions. Any settings make on this tab are specific to the current batch in the job only. They will not apply to existing batches. Job properties are stored in a batch file header at the time the batch is created.

To set these options:

1. Open a job within Designer and display the Job Properties, Options tab for the job.

2. In the **Auto Options** area, you can select from the following check boxes:

   o **On when error** - Leaves the auto function on when an error message is received. This allows the operator to advance to the next field with rejects. By default, when an auto error condition is encountered during a Completion session, the auto function is turned off and remains off until the operator presses [F2].

   o **Off when back** - Turns the auto function off when a record back condition occurs. By default, when an operator moves back a record by pressing the [PgUp] key, the auto function remains on, disabling an operator from moving back to the data file to make changes or view data. Use this option if operators need to move back in a data file.

   o **On when forward**- Leaves the auto function on when a record or field forward condition occurs. A record or field forward condition is called when the operator presses either the directional or Record Forward [PgDn] keys. This allows the operator to continue normal keying, such as advancing to fields and rejects. By default, when an operator moves forward a record, the auto function toggles off.

   o **On when location**- Turns the auto function on only when the operator reaches the location field, or the furthest point in the batch.

3. In the **Error Options** area, select one of the following radio buttons to control what happens when an error message displays during Completion:

   o **None** - No additional behavior occurs when an error message is encountered during a Completion session.

   o **Beep** - A beep sounds when an error occurs during a Completion session.

   o **Tone** - A tone sounds when an error occurs during a Completion session.

   o **Sound** - The system sound occurs when an error occurs during a Completion session.

   o **Flash** - The computer screen flashes when an error message displays during Completion.

4. In the **Oversign Options** area, select from the following options to distinguish between positive (unsigned) and negative (signed) numeric values. Many backend or mainframe systems cannot accept a minus sign to indicate negative values. Instead, a range of alpha characters is used to indicate a negative number. For example, a 1 converts to a J, a 2 converts to a K, up to 9 which converts to an R. 0 converts to an ampersand (&), but can be redefined to coincide with the constraints of the user's system.

   o **After character** - Change the function of the Oversign key [F8] to be pressed after the last character. The default setting is for the Oversign key [F8] to be pressed before the last character.

   o **No release** - Prevents release of the data field once the operator presses the Oversign key [F8]. If this field is selected, no field release is performed after the Oversign key is pressed. This allows the operator to oversign multiple characters within the same field.

   o **0 character** - Specifies another character (other than the default) for oversign. The default oversign character for 0 is "&". Some mainframes use a value other than an ampersand to denote a negative number ending in 0.

5. Select the **Non destructive field release** check box to save the data in the field to the right of the cursor when the operator presses [Enter]. The default behavior is for the data in the field

to the right of the cursor to be deleted when an operator presses [Enter] during a Completion session.

6. Select the **Replay field flags** check box to check the batch for error flags when the operator closes the batch data file during a Remove Flags pass of Completion. If the batch contains error flags, the batch is kept open and the cursor is positioned on the first error flag in the batch.

7. Select the **Auto close on last record** check box to automatically close the batch when all fields and records in the batch are processed. If the operator is using JobFlow and this option is on, the next batch in the JobFlow task is loaded. This option is stored in the batch data file. Changing this option for a job will not change it for a batch within an existing data file. See also **Auto close in verify** option.

8. Select the **Record insert in update** check box to disable the location field functions when the operator inserts a record during the Update mode of Completion. Set this option so that the location field functions are performed when a record is inserted in Update mode.

9. In the **Idle time before close** field, select the amount of time (in seconds) to lapse before the batch is automatically closed or interrupted and the user logged out of the JobFlow.

10. Select the **Link performs fmtcor** check box to have the LINK verb in an FPL Field Edit perform a format correct on the current location, rather than waiting for record release. By default, the LINK verb is not invoked until the current record is released.

11. Select the **Auto close in verify** check box to close the batch when all fields and records in the batch data file are verified. If the operator is using JobFlow and this option is on, the next available batch is loaded in Completion. See also **Auto close on last record** option.

12. Select the **Close at anytime** check box to allow the operator to close an incomplete batch before all the images are processed. This option allows you to force an edit to run on an application-specific level. Use this option with caution as it can lead to exporting partial files.

13. Select the **Location return in update** check box to return the operator to the location record rather than to the beginning of the batch when a batch is interrupted and then resumes during the Update mode of Completion.

14. Select the **Display form list in manual ID** check box to display the Change Form dialog during Manual Form ID, which is the Identify Images mode of Completion. If you do not select this option, the operator must select the **Record > Change Form** menu item in the Completion module to display the Change Form dialog.

15. Select the **O29 on** check box to turn on the 029 keyboard function, which is a numeric keypad embedded in the standard, alphabetic keyboard.

16. Select the **Dialog on** check box to enable the basic function keys during Completion to mimic the standard input behavior of those in Windows. The following functions apply when you select this option:

    o   **[Tab]** - Moves to the next field, autowraps

    o   **[Back Tab]** - Moves to the previous field, autowraps

    o   **[Backspace]** - Destructive backspace function

    o   **[Page Down]** - Moves to the last field, then to the next page

    o   **[Page Up]** - Moves to the first field, then to the previous page

    o   **Single Mouse Click** - Moves to any field on the form without an "End of Data" error.

17. Select the **Browse** button adjacent to the **Keyboard map** field to specify a keyboard map file to use for this job. See the Mapping the Keyboard section for more information.

18. Select the **Save All** button on the Designer window toolbar to save your changes.

Setting Display Properties for the Job

Use the options on the Job Properties, Display tab to set display properties for fields and labels. These settings are reflected in *Completion* (not in Designer). To set these values:

1. Open the job in Designer.

2. Display the Job Properties, Display tab for the job.

3. Select from the following check box options:

   o **Field labels** - Display all field labels to the operator during a Completion session (this is the default). De-select this option to hide field labels during Completion.

   o **Field edit changes** - Display FPL Field Edit changes immediately, rather than when the record is released (i.e., when a Field Edit program moves values to field(s) before to the location field).

   o **Display all fields** - All recognition data is read into the current record and is visible to the operator in any Completion mode. Selecting this option, disables blind verification during Completion.

   o **Display field snippets** - Create and display field snippets during a Completion session based on the data field recognition zones. Each field snippet appears on the data form directly above the associated field.

   o **Recognition data** - Displays all of the recognition data during the Entry mode of Completion, but also allows for blind verification. The operator is positioned on the first reject.

   o **Maintain record display** - Maintain the display of the data on the data form when the operator performs a record write and then begins typing data during the Verify mode of Completion.

   o **Scroll by page** - Scroll the entire data form a page at a time if not all of the fields are visible. The default behavior is to scroll a line at a time.

   o **Hide image display** - Hide the image viewport display during a Completion session.

4. Select **Save All** on the Designer window toolbar to save your settings.

Setting Aux Dup Properties

The Job Properties, Aux Dup (auxiliary duplication) tab allows you to display the Aux Dup table for the current job within a FormWare text editor. The Aux Dup table contains a list of common values that an operator can insert into fields during a Completion session. Here, you can edit the Aux Dup table by adding, changing, or deleting text values. The Aux Dup table is stored as a file with a .aux extension in the \FormWare\Jobs folder.

During a Completion session, if the operator presses the Aux Dup key (the default is the [Home] key), the Aux Dup table displays. The Completion operator can then select the desired item either with the mouse or with the [↓] key and then press [Enter]. The item selected is inserted into the data field and the cursor is positioned immediately after the last character of the inserted data.

To set the Aux Dup properties:

1. Open the job in Designer.

2. Display the Job Properties, AuxDup tab for the job.

3. Select the **Text** button to display the Aux Dup table for the current job. If field values already exist in the Aux Dup table, they will appear in the text editor.

4. Type the table values you want. You must place each value on a separate line followed by a carriage return.

5. Select **Save** on the text editor toolbar and **File > Exit** to exit the text editor.

6. Select **Save All** on the Designer window toolbar to save your settings.

Determining Completion Settings

Understanding the Modes of Completion

There are several modes for the data Completion of batches. Each mode performs a different part of the forms processing solution. Access these modes through the File menu in the Completion window or through the Completion Processing dialog (accessed by selecting the **Process > Completion Processing** menu option in Administration). An explanation of each mode follows.

- **New** - Starts a new batch for reject repair or data entry from image or paper.

- **Append** - Opens an existing batch and enables new records to be processed at the end of the batch.

- **Verify** - Opens the batch for verification. The verification option is set on each field when created in the Designer module.

- **Correct** - Opens the batch in Correct mode. This is a visual verification mode. The operator can press either the [Enter] key or [Down Arrow] key to move from field to field. To change any data, the operator must press a correction key.

- **Update** - Opens the batch in Update mode. This mode is similar to Verify mode, however, the cursor stops only on fields that contain the update indicator. The update option is set for the field in the Designer module.

- **Remove Flags** - Opens the batch for flag removal. The program searches for error flags to repair. When the program finds a flag, it positions the cursor in the first position of the field. The cursor is not placed on the error flag unless the flag is in the first position of the field.

- **Quality Check** - Opens a batch for a quality check. Quality Check mode enables the operator to verify every n^{th} record or the specified record. You must set the QCR parameter in the job INI file for this option. The option uses the same verification option as Verify mode.

- **Read Only** - Opens the batch in Read Only mode. The operator can press either the [Enter] key or [Down Arrow] key to move from field to field. Changes cannot be made in Read-Only mode.

- **Add New Images** - Appends new images to an existing batch. If you add new images, you must place the new image names after all previously processed images in the BDF. The program compares the image list stored with the batch to the image list in the BDF. If it finds a discrepancy, it adds the new images and positions the cursor on the first appended image record for processing.

- **Identify Images** - Processes the images that failed the form identification process. When the operator starts a batch in this mode, each image that failed is presented for

identification. The operator identifies the form's number and moves to the next image. When the operator identifies the form, the BDF is updated with the form number. The batch can then be routed to recognition or reject repair for further processing. A custom menu is available when performing manual form identification in this mode.

- **Image Quality Check/Rescan** - Displays the batch images in thumbnails for the scanning operator to view and reject any that were not scanned properly. The invalid images can be marked for rescanning, or if a scanner is attached to the workstation, the images can be rescanned at that time.

- **Batch Build** - Pre-builds FormWare batch files. Batch Build takes a list of images in a BDF and their associated OCR data and creates a batch data file. Using a command line parameter with the Batch Build task, it is possible to return a custom return code that will route that batch past the Completion – New task so the batch will not be presented to an operator for keying when no error characters exist.

Opening Batches in Completion

The Administration module is a starting point for the FormWare modules.

Creating a New Batch in Completion

To create a new batch within the Completion module, follow these steps.

1. From the Administration window, select the Completion button on the toolbar. The Select job for new file dialog displays.

2. Highlight a job to associate with the batch and select **Open**. The properties set in this job are then applied to the newly created batch. This means that the data form(s) in the selected job display with the batch images and all other properties (e.g., field edits, file edits, or export programs) are applied to the batch data. The Enter New File Name dialog displays.

3. In the **File name** field, type a batch name and select **OK**. This dialog only accepts A-Z, a-z, 0-9, "_" and "–" as valid characters. A batch data file is then created with this name. To associate the batch with an existing BDF, the batch name must be the same as the BDF name and without an extension. If you type a batch name that does not match an existing BDF, the batch will not be associated with the data file. Additionally, FormWare will not allow you to give your data file a name that has already been used by another data file.

   The main Completion window displays the first record with a data form from the selected job and the images from the associated BDF. If recognition has been performed on the BDF image files, then recognition results also display.

4. You are now ready to begin entering data.

5. To exit Completion, select the **File > Exit** menu option when the batch is complete, or select **File > Interrupt**.

Opening Batch Data Files in Completion

To open an existing batch within the Completion module, follow these steps.

1. From the Administration window, select the **Completion** toolbar button.

2. Close the Select job for new file dialog.

3. Select a Completion mode from the File menu. The Select file to (Completion Mode) dialog displays the contents of the \FormWare\Files folder.

4. Highlight a batch data file and select **Open**. Batch data files display without any file extension, and BDF files are saved with a .bdf extension. The main Completion window displays with the selected batch file information, data form, and images in the mode you selected. This mode displays on the status bar, in the lower left-hand corner of the Completion window

    You are now ready to conduct a Completion session.

5. To exit Completion, select the **File > Exit** menu option when the batch is complete, or select **File > Interrupt**.

Opening Existing Batches in Administration

If you want to open an existing batch in the Completion module from the Administration module, follow these steps:

1. From the Administration window module.

2. Display the contents of the Files folder in the File Browser.

3. Select the **Show All Files** button on the toolbar to view all files in this folder.

4. Double-click on a batch data file. This file has the base name of a BDF, but does not have an extension. The Completion window displays the batch data file information with the associated data form and images in Correct mode.

5. You are now ready to conduct a Completion - Correct session.

6. To exit Completion, select the **File > Exit** menu option when the batch is complete, or select **File > Interrupt**.

Opening Completion from the JobFlow

The optimal way to use the Completion module is with a JobFlow system. Completion is designed to integrate seamlessly into a JobFlow system and assign Completion operators batches to process. To perform Completion using a JobFlow system, the JobFlow Client must be installed on the system. When you assign batches to the Completion module through a JobFlow system, the operator is presented with the batch in a mode specified by the JobFlow.

Take the following steps to access the Completion module from JobFlow. This information presumes that a JobFlow system is already in place and that a batch is available for Completion processing.

1. From the Administration window, select the **JobFlow Client** button on the toolbar. The JobFlow Client dialog displays.

2. In the **Job Flow** field, select an existing JobFlow from the list. The Task field then populates with the tasks assigned to the JobFlow.

3. Select a Completion task from the **Task** field.

4. Select **OK**. This launches the Completion module in the selected mode with the currently available batch. If a batch is not currently assigned or available for the JobFlow task, you receive a prompt informing you of this.

5. You are now ready to conduct the selected Completion session.

6. To exit Completion, select **File > Exit** when the batch is completed or select **File > Interrupt**.

Automating the JobFlow Process

Instead of requiring an operator to follow the steps above to access a batch, you can automate the process. To do this, create a shortcut on the workstation desktop and assign specific properties to it. This way an operator only has to double-click on the shortcut to be logged into the assigned JobFlow task. To create a shortcut:

1. On the Completion workstation, right-click on the desktop and select **New > Shortcut** from the context menu.

2. A wizard dialog displays to help you create the shortcut. When it prompts you for the command line or location of the shortcut item, type:

   ```
   \FWShare\FormWare\Programs\JFCLIENT.EXE
   ```

3. After the program name, also type:

   ```
   /b=%BCH% /f=<JobFlow name> /n=<JobFlow task number>
   ```

4. The /b switch is optional.

5. Type a name for the shortcut.

6. Select **Finish** to create the shortcut.

Setting Up the Attachment Viewer

Attachments are images that are used during a Completion session, but not for extracting data. They do not have a record associated with them. Instead, they contain supplementary data that is not extracted using a recognition engine. FormWare allows you to view attachments during Completion. You set options for the attachment viewer in the Designer module. To set up the attachment viewer:

1. Open a job in Designer.

2. Display the Job Properties, Image tab for the job.

3. In the **Attachment Viewer** area, select the **Display** check box to open the attachment window when the Completion session is initiated.

4. Select the **Include current image** check box to designate the image in the Completion window as the first image to appear in the attachment window. If this option is not checked, the first image displayed will be the first attachment.

Note: Since scale to gray properties are always applied when viewing images in an attachment window, selecting the **Scale to gray** check box will not change the image view.

5. In the **Height** field, select the height in pixels for the attachment viewer to use to display images when the Switch Views feature is used.

6. In the **Width** field, select the horizontal pixel count the attachment window uses to display the image when the view of the attachment is set to the side-by-side display. The side-by-side view is the default view of the attachment window.

7. In the **Scroll** field, select the pixel count the image will scroll when the up and down arrows are used in the attachment window. 100 pixels is the maximum.

Note: You can also add the VSM = # INI parameter to the System.ini or the job INI file to enable scrolling values greater than 100. Replace # with the number by which you want the scroll value to be multiplied. For example, if you set the **Scroll** value to 75 in Designer and you set the parameter VSM=3 in the job INI file, the attachment scrolls by 75 x 3 = 225 pixels. The VSM parameter applies only when the embedded Attachment Viewer is used to display attachments in Completion.

8. In the **Scale** field, select the scale factor you want to use as the default in the attachment viewer. When the scale factor is set, the on-screen image becomes X% of the original size of the image.

9. In the **Rotate** field, select the rotation factor you want to use as the default in the attachment viewer. When the rotation factor is set, the image will rotate X degrees from its original position.

10. Select the **Save All** button on the Designer window toolbar to save your settings.

Setting Keyboard Map Values for the Attachment Viewer

The Attachment Viewer has keyboard map values that can be accessed while capturing data in Completion. The keyboard map values can also be used by operators as *Pushkey* functions when performing high-speed recognition reject repair during Completion. It is often faster for operators to use *Pushkey* functions than it is for them to use menus or toolbars. The keys are mapped using the function numbers listed in the chart below.

Note: Previously, when you enabled the attachment viewer in Completion, you had to manually add the attachment viewer control keys to a keyboard map file to use them. Completion will now automatically map the control keys for you when you activate the attachment viewer and have not previous mapped the control keys in the keyboard map file.

Attachment Viewer Keyboard Function	Keyboard Value	Description
Attachment Viewer/Image Viewport Toggle	969	Map this value to a key to toggle between views of the Image Viewport and the Attachment Viewer. To use this keyboard map value, first use either the Minimize Attachment Viewer function or the Minimize the Image Viewport function. Then, when either the Attachment Viewer or Image Viewport window is minimized, use the Attachment Viewer/Image Viewport Toggle function to switch between the Attachment Viewer and the Image Viewport. Only one image is displayed at a time.
Scroll Image Left	970	Map a scroll image value to a key to scroll the attachment image left, right, up, or down. You set the amount the image scrolls in Designer. To set the number of pixels an image scrolls, type a number in the Scroll field in the Attachment Viewer group on the Image page. The maximum acceptable value in the Scroll field is 100 pixels. You can set scroll values greater than 100 by using the VSM = # INI parameter.
Scroll Image Up	971	
Scroll Image Right	972	
Scroll Image Down	973	
View Previous Image Within Document	974	Map this value to a key to move to the previous image within the attachment document. If you select this value when you are viewing the first page of the attachment, a message displays indicating that you are viewing the first page.

Attachment Viewer Keyboard Function	Keyboard Value	Description
View Next Image Within Document	975	Map this value to a key to move to the next image within the attachment document.
Move to Top of Attachment Image	976	Map this value to a key to move to the top of an image in the Attachment Viewer in Completion.
Move to Bottom of Attachment Image	977	Map this value to a key to move to the bottom of an image in the Attachment Viewer in Completion.
Rotate Image Left	978	Map one of the rotate image values to a key to rotate the attachment image left, right, or 180 degrees. The Rotate Image Left and Rotate Image Right values rotate the image 90 degrees in the selected direction.
Rotate Image Right	979	
Rotate Image 180 degrees	980	
Scale to 10%	981	Map one of the scale to X% values to a key to change the scale of your image. When one of these values is selected, the image becomes X% of the original size of the image.
Scale to 25%	982	
Scale to 33%	983	
Scale to 40%	984	
Scale to 50%	985	
Scale to 66%	986	
Scale to 75%	987	
Scale to 80%	988	
Scale to 90%	989	
Scale to 100%	990	
Scale to fit width	991	Map this value to a key to change the scale of your image to fit within the width of the Attachment Viewer window. When you use this value, the width of your image fits perfectly within the width of the Attachment Viewer window.
Switch attachment view window	993	Map the Switch Views to a key to change the placement of the attachment in the display window. The default view places the attachment and the field data side-by-side, with the attachment on the left. The size of the attachment window is determined by the Width value set on the Job Properties, Image tab. Using the Switch Views function, you can switch the placement of the attachment to the top of the window by setting the Height instead of the Width. The top attachment window size is determined by the value in the Height property on the Job Properties, Image tab. This function also enables you to minimize the attachment window when the Switch View key is pressed. To minimize the attachment window, set the Height property value to less than four on the Job Properties, Image tab.
Display First Attachment	997	Map this value to a key to move to the first attachment when the Attachment Viewer displays an image beyond the first in the series. Using this value with the keyboard mapping values 974 and 975 allows movement to previous and next attachments. If you select Include Current Image on the Job Properties, Image tab, the first attachment is the same as the image in the Image Viewport.

Attachment Viewer Keyboard Function	Keyboard Value	Description
Minimize Attachment Viewer	998	Map the keyboard value Minimize Attachment Viewer to minimize the Attachment Viewer and use the entire screen to view the Image Viewport and data form when in Completion. Conversely, map the keyboard value Minimize Image Viewport to minimize the Image Viewport and use the entire screen to view the Attachment Viewer and data form when in Completion. Use these keyboard map values during the default view of the Attachment Viewer or after performing a switch of the image view.
Minimize Image Viewport	999	

Use the map values listed in the following table to map the Attachment Viewer keyboard functions to keys on the keyboard. All Attachment Viewer functions are mapped to the [Ctrl] key plus another key. For example, to map the Scroll Image Left function to the [Ctrl] key plus the left arrow key, add the equation CK25=970 to the Keyboard.ini file.

Control Key + Key	Map Value
Ctrl + left arrow	CK25
Ctrl + up arrow	CK26
Ctrl + right arrow	CK27
Ctrl + down arrow	CK28
Ctrl + Page up	CK21
Ctrl + Page down	CK22
Ctrl + any alpha	CK_X (where X is any alpha character)

Following are sample values to enter in the Keyboard.ini file for the Attachment Viewer keystroke functions:

INI Parameter	Attachment Viewer Function
CK25=970	/* Left = Scroll Left */
CK26=971	/* Up = Scroll Up */
CK27=972	/* Rght = Scroll Right */
CK28=973	/* Down = Scroll Down */
CK21=974	/* PgUp = Previous Image */
CK22=975	/* PgDn = Next Image */
CK_L=978	/* 'L' = Rotate Left */
CK_R=979	/* 'R' = Rotate Right */
CK_F=980	/* 'F' = Flip Image */
CK_1=981	/* Scale image to 10 % */
CK_2=982	/* Scale image to 25 % */
CK_3=983	/* Scale image to 33 % */
CK_4=984	/* Scale image to 40 % */
CK_5=985	/* Scale image to 50 % */
CK_6=986	/* Scale image to 66 % */

INI Parameter	Attachment Viewer Function
CK_7=987	/* Scale image to 75 % */
CK_8=988	/* Scale image to 80 % */
CK_9=989	/* Scale image to 90 % */
CK_0=990	/* Scale image to 100 % */
CK_W=991	/* Scale to fit width */
CK_V=993	/* Switch Views */

Setting Image Quality Check Properties for Completion

Perform the following steps to determine how the Image Quality Check (IQC) process will display in Completion:

1. Open a job in Designer.

2. Display the Job Properties, Image tab for the job.

3. In the **Image Quality Check** area, select one of the following options:
    - o **Full size** - Displays thumbnail images at full size in Completion.
    - o **Fit to width** - Displays thumbnail images at the maximum width of the image viewer in Completion.
    - o **Fit to height** - Displays thumbnail images at the maximum height of the image viewer in Completion.

4. Select **Save All** on the Designer window toolbar to save your settings.

Defining INI Settings for Completion

You can use INI parameters to add or change program functionality during a Completion session. Within FormWare, INI parameters can control or impact functionality at a high level or a low level, depending on the INI file where the parameter is set. For example, setting a Completion parameter in the System.ini file affects all Completion sessions on the system. Setting a parameter in a <job name>.ini file affects all Completion sessions within the specified job.

You can also use the "-INI=<filename>.ini" client parameter to control program functionality at Completion session task level. See Disabling Display of Menus and Menu Items with a Completion Session that follows for more information.

For a list of the available INI parameters, see INI Parameters in the Supplementary Information section.

See the Setting INI Parameters section for more information on setting INI parameters within FormWare.

Note: Each INI parameter must be separated by a <crlf> (carriage return/line feed combination) within the INI file. There must also be an extra <crlf> at the end of each INI file.

Disabling Display of Menus and Menu Items within a Completion Session

Use the INI parameter DMU= to disable the display of menus and menu items on the Completion window. Performing this task enables you to control the display of only those menus and menu

items that are used by an operator during a Completion session. The steps in this task instruct you on how to create a unique INI file and then apply that INI file to a specific Completion task, although the DMU= parameter can also be applied at the system (System.ini) level within FormWare.

1. On the Administration window, display the contents of the FormWare\Ini\System folder in the File Browser. (Select the Show All Files button to display all files in this folder.)

2. Right-click in the Contents area and select New from the context menu.

3. Type DMU parameters as follows:

   ```
   DMU=File:Append;Remove Flags;Identify Images
   ```

   In this example, the File is the menu containing menu items – Append, Remove Flags, and Identify Images – that you want to remove from the Completion menu bar. Delimit the menu name with a colon and each menu item with a semi-colon.

4. When you use the DMU parameter with only a menu name, for example:

   ```
   DMU=File:
   ```

   Removes the entire File menu from the menu bar.

5. Add a <crlf> at the end of each DMU line within the INI file.

6. Insert a <crlf> at the end of the INI file. Here is a INI file example:
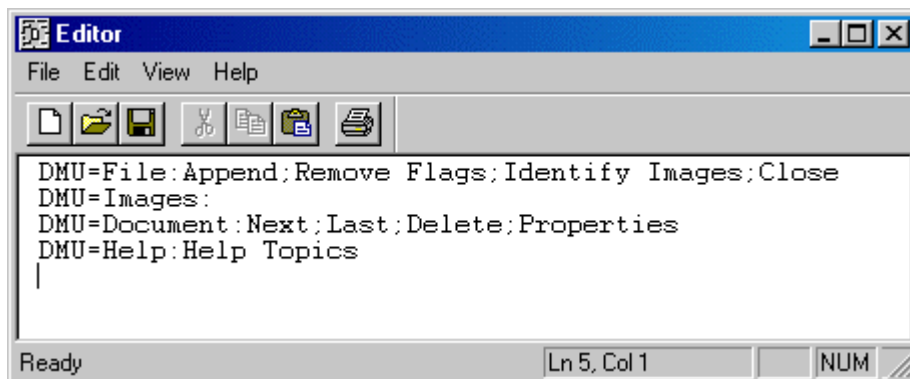
7. Select **File > Save** and save the INI file with a meaningful name (e.g., CMP_DMU.ini).

8. Close the Editor and return to Administration.

9. Set up a Completion task to use the INI=<filename>.ini parameter. This can be done in one of two ways:

   a. As a command line argument. For example:

      ```
      complete -e -INI=CMP_DMU.ini
      ```

      This command runs Completion – New mode with no image or form. The menus and menu items you defined in your INI file will not appear on the Completion window menu bar.

   b. As a JobFlow task in JobFlow Designer with -INI=<filename>.ini as a client parameter. For example:
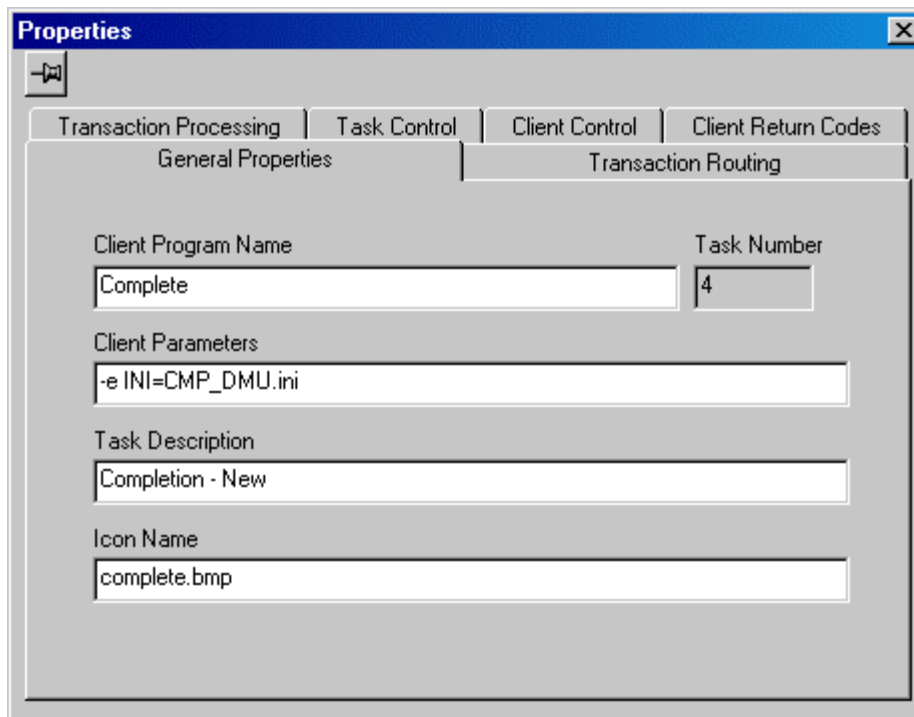
To access this Task Properties dialog, select the Completion task in the JobFlow Browser of the JobFlow Designer window.

For more information on the DMU= parameter, see INI Parameters, Completion INI Parameters in the Supplementary Information section.

Changing the Viewport Background Color

You can change the background color for the empty area in the image viewport for Completion. Add the BIC= INI parameter to the System.ini or job INI file to supply the RGB color selection for this area. This parameter requires the hex equivalents of the red, green, and blue values. For example, the BIC= INI setting for a light gray background would be BIC=14474460.

Skipping Rejected Records During Completion

Use the FWSKIPONREJECT INI parameter to enable Completion to skip previously rejected records. For example:

```
FWSKIPONREJECT=xyz
```

where the xyz is a string value that represents one or more reject code characters.

When this parameter is invoked and a previously rejected record has a reject code that matches one of the characters in this string, the rejected record is skipped. If the asterisk character (*) appears in the string, then any reject code character will cause the rejected record to be skipped.

You can set the FWSKIPONREJECT= parameter in a variety of INI locations depending upon where this functionality needs to be applied. Insert the parameter within the System.ini file for system-wide application; within a client INI file to apply within instances of a particular client; as a workstation environment variable to apply to all clients on a particular machine; or as a command line argument to apply within an instance of a particular WorkFlow task or step.

Establishing Dynamic Form Loading

Use the Dynamic Form Loading INI setting, DFL=Y, to control how forms are loaded in Designer and Completion. If you set DFL=Y in the System.ini or a job INI file, the forms for a job are only loaded into memory when the application needs those forms. Using this parameter can decrease the amount of time necessary to load an application into memory when using Completion or Designer. Consider using this setting when you are designing or running job that contains a large number of forms.

Creating Indexes

You use indexes to locate data within a batch. You can create an unlimited number of indexes for the same batch. Indexing the batch using a specified key creates an index. The Index folder contains a list of all indexes in the FormWare root directory.

A key is a field, variable, or literal in the batch data file on which the index is created. For example, in a record that has City, State, and Zip Code fields, you may want the index based on the states. The State field would be the index key.

When a batch data file is indexed, each index points to the location of the indexed record in the sequential file. Such pointers facilitate searching for data in the batch, regardless of the size of the indexed batch.

To create an index used as a value table during a Completion session or a look-up table used during a program, you must have a valid FormWare batch. Complete the following steps to create a FormWare index.

1. Open a job in Designer.

2. Display the Job Properties, Export tab.

3. In the **Export Device** field, select the adjacent **Browse** button to locate an existing export device in the \FormWare\Devices folder. Or, type a new device name in the field and select **Edit** to edit the new device.

4. On the Designer window in the data form area, create a form with all of the data fields for the necessary information.

5. Create a batch data file with your job. The batch can be created by an operator who enters data into the batch during Completion or with the batch import functions in the Administration module.

6. Create a sort program to organize the data in your batch.

7. Execute the sort program against your batch. Assess your index for correctness. If correct, the index is now ready to use.

To view an index file:

1. From the Administration window, display the contents of the \FormWare\Index folder in the File Browser.

2. Right-click on an index file in the Contents area to display the context menu.

3. Select **Send to > Index Data Viewer**. The editor displays each indexed field (called a key), the file in which it belongs, and the offset or pointer to the batch.

Configuring a Document Assembly Application

Document Assembly enables you to design a FormWare application for the purpose of verifying the correct sequence and completeness of a collection of documents. This is useful in industries that process collections of documents (e.g., mortgage loan applications), where the order and completeness of the documents takes priority over data indexing or data capture of the document contents. This type of application can also be used for manual organization and assembly of a batch of materials prior to data entry. For these reasons, Document Assembly is performed as a Completion – New task within FormWare, which occurs prior to any data entry process within the application.

Note: While the Image QC/Rescan mode (Manual IQC) does provide the ability to perform basic document assembly functions, its main focus is to review/correct image quality or to perform high-speed Manual Form ID. When your business requirement is more focused on actual document assembly, then the Document Assembly Completion task should be used.

Understanding the Business Problem

The ABC Company is in the business of processing mortgage loan documents. The main ABC facility receives a pallet of boxes from a mortgage company. Each box has a bar code on the front for identification. Within each box is a collection of large file folders. Each file folder contains the documents for one mortgage loan and also includes a bar code for identification. Based on the loan type there are an expected number of documents sorted in a specific order within each file folder. Along with the crate of boxes is an electronic file that identifies the boxes included with the shipment as well as the file folders within each box. This file is generated by scanning the bar codes on each box and folder as they are prepped for delivery to the main facility. Index values for all documents are based on this electronic file. The documents are prepped for scanning and delivered to the scan operator. Once the operator scans the documents, how does the company:

- Verify that the contents of each box and folder are complete and in the correct order?
- Verify that breaks in each document have been correctly identified?
- Verify that the pages of each document are in the correct order?
- Verify that the quality of each image is adequate for storage and retrieval and rescan it if is not?
- Verify that the index value associated with each image is correct?
- Change the order of documents within a collection or images within a document?
- Insert or delete images within a document or documents within a collection?
- Rescan or rotate an image?
- Modify or key the index values for an image?

Understanding the Business Solution

Once all documents within the boxes are prepped for scanning, they are delivered to a scan operator and scanned into the FormWare system. Each box equals a Level 4 data item. Each folder within the box is a Level 3 data item. Each document in a folder is a Level 2 data item and each image within a document is a Level 1 data item. (See the FWLEVn= Parameter section of Understanding Document Assembly BDF Parameters for more information.)

Once all documents are scanned, a VBA edit task is run to merge the batch of documents with the shipment's electronic file, which contains the appropriate index values for each document. The

document type is also used as an index value. These index values are assigned to an image using APD parameter during this step. (See the APD= Parameter section of Understanding Document Assembly BDF Parameters for more information.) Because each loan type contains a certain number of documents that must be included, and must be in a pre-defined order, an assumption is made that the documents that were scanned match the pre-defined list of document types and their associated index values from the electronic file. To verify that this is indeed correct, the batch is sent to Document Assembly where an operator verifies that the batch is in the correct sequence and that each document is assembled correctly within that batch. The operator also reviews image quality at this time. Document Assembly enables Completion data entry operators to:

- Verify the contents of a batch are complete
- Verify the contents of a batch are in the correct order
- Verify the document breaks have been identified correctly
- Verify the pages of a document are in the correct order
- Verify the image quality is sufficient for storage and retrieval
- Verify the index values associated with an image are correct
- Modify the order of documents within a batch
- Modify the order of images within a document
- Insert or delete an image within a document
- Insert or delete a document within a batch
- Rescan or rotate an image
- Modify or key the index values for an image

Understanding Document Assembly Client Parameters

To run Completion – New mode as a Document Assembly task, you must include the following client parameter:

```
–BLD=Y|B[D]
```

The –BLD= parameter pre-builds the specified batch or batches for Document Assembly. This parameter can be passed as a client parameter for a task within JobFlow Designer or as a command line parameter. Valid parameter values are:

- **Y** – Instructs Completion to pre-build the batch with OCR data and with property values within the BDF. This option requires more processing overhead than the –BLD=B parameter because it not only reads the information from the BDF file and pre-builds the batch with OCR data (similar to a Batch Build task).

- **B** – Instructs Completion to pre-build the batch with property values from the BDF only. This assumes that each image has an APD property set and the contents of the APD value will be filled into the fields of the form during the pre-build function. Use the –BLD=B parameter when you are running an application that has performs Document Assembly functions exclusively.

- **D** – (Optional) Instructs Completion to delete the batch data file at the end of the Completion session. This option is useful when you are performing repeated tests of the Document Assembly process and can be run in either 'B' (–BLD=BD) or 'Y' (–BLD=YD) mode.

Understanding Document Assembly BDF Parameters

Within a Document Assembly runtime environment, the BDF is generated as part of a pre-Completion edit task using VBA or another similar tool to generate the BDF file with the correct values. For testing purposes, you can manually create a BDF file within FormWare system and use it to test processes prior to production. See Testing the Document Assembly Job for instructions on how to create a test BDF for Document Assembly.

Here is an example of a BDF for Document Assembly:

```
JOBNAME=DOCASSMBLY
FWLEV4=BOX00000001
FWLEV3=LOAN0000000001
FWLEV2=01
APD=BOX00000001 LOAN0000000001 01 1
01:\fwshare\formware\Images\DEMO0001.tif
APD=BOX00000001 LOAN0000000001 01 2
01:\fwshare\formware\Images\DEMO0002.tif
APD=BOX00000001 LOAN0000000001 01 3
01:\fwshare\formware\Images\DEMO0003.tif
APD=BOX00000001 LOAN0000000001 01 4
01:\fwshare\formware\Images\DEMO0004.tif
FWLEV2=*
APD=BOX00000001 LOAN0000000001 01 5
01:\fwshare\formware\Images\DEMO0005.tif
APD=BOX00000001 LOAN0000000001 01 6
01:\fwshare\formware\Images\DEMO0006.tif
APD=BOX00000001 LOAN0000000001 01 7
01:\fwshare\formware\Images\DEMO0007.tif
APD=BOX00000001 LOAN0000000001 01 8
01:\fwshare\formware\Images\DEMO0008.tif
APD=BOX00000001 LOAN0000000001 01 9
01:\fwshare\formware\Images\DEMO0009.tif
APD=BOX00000001 LOAN0000000001 01 10
01:\fwshare\formware\Images\DEMO0010.tif
FWLEV2=02
APD=BOX00000001 LOAN0000000001 02 1
01:\fwshare\formware\Images\DEMO0002.tif
```

FWLEVn= Parameter

The FWLEVn= parameter identifies the data item level within the FormWare. Up to eight levels can be defined within FormWare to create hierarchical classifications of data that passes through the system. Within a Document Assembly application, defining the level hierarchy is critical to successful processing. The BDF contains all the levels that will be used in the Document Assembly application. For example:

```
FWLEV4=BOX0000001
```

Indicates that a Level 4 data item is a box of documents. All FWLEVn= parameters for a particular batch must be "declared" in descending sequential order at the top of the BDF file beneath the JOBNAME= parameter.

APD= Parameter

The APD= parameter (Add Page Data) sequentially inserts the level information from the FWLEVn= parameters into the batch file record for the image immediately following this parameter. By creating a corresponding field on the data form for each level value stated in the APD parameter, you can verify whether the appropriate hierarchy is assigned to each image in the batch. For example:

```
APD=BOX00000001 LOAN0000000001 01 1
01:\fwshare\formware\images\DEMO0001.tif
```

Attaches level information to the DEMO0001.tif image.

Note: The APD values are applied during the pre-build function to fill the fields of each record. These values are then stored as part of the batch data file. If you make changes to these fields on the data form during Completion, the changes are reflected in the batch data file and not in the original APD values within the BDF file.

Understanding Document Assembly Job INI Parameters

A number of INI parameters, set at the job level, can be used within Document Assembly applications. These INI parameters work in concert with the BDF file parameters and client parameters to determine how various parts of the Document Assembly job will function. To use these parameters, you must create a job INI file in the \FormWare\Ini\Job folder that has the same name as your Document Assembly job.

Here is an example of a job INI file for Document Assembly:
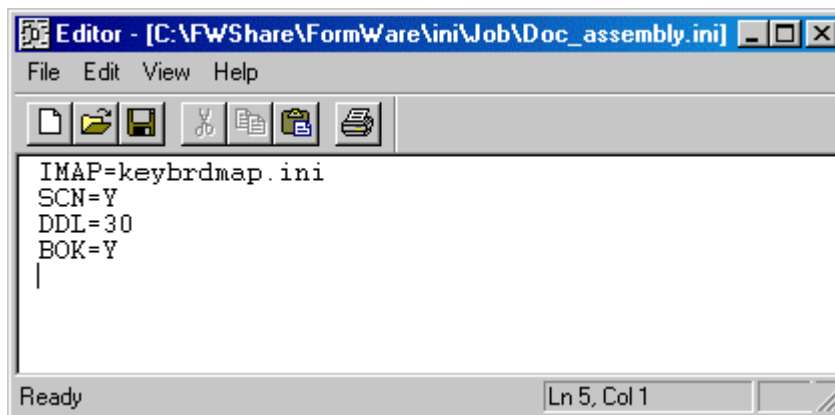
BOK= Parameter

The BOK=Y parameter (Browse OK) enables the **Browse** button on the Image Insert dialog within Completion. Select the **Browse** button to access the Open dialog, which enables you to browse outside of the default image folder for images to insert.

On the Image Insert dialog, you can type the name of an image in the **Image Name** field or select one from the list. Here is an example of the Image Insert dialog:

Image Insert

Image Name

[   ]   [OK]

                                 [Cancel]

```
00000001B.TIF
1.OCR
1.TIF
10.OCR
10.TIF
11.OCR
11.TIF
12.OCR
12.TIF
13.OCR
13.TIF
14.OCR
14.TIF
15.OCR
15.TIF
16.OCR
16.TIF
17.OCR
17.TIF
18.OCR
18.TIF
19.OCR
19.TIF
2.OCR
```

[Browse]   [Scan]
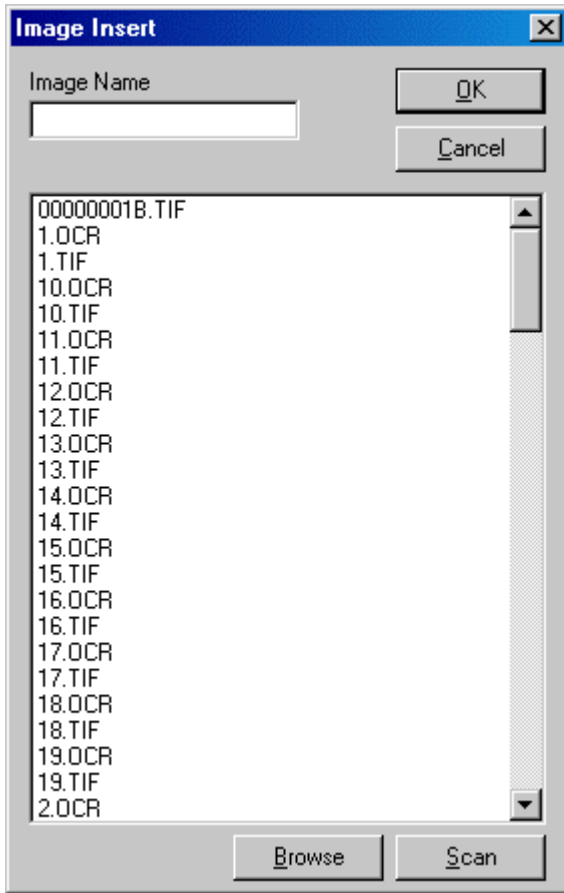
DDL= Parameter

The DDL= parameter (Display Data Length) enables you to include a portion of the record data after each image name within the Data Items dialog hierarchy when the document level is expanded. This can be useful to help the Completion operator identify the record. For example:

    DDL=30

Displays the first thirty characters of data from the record immediately after the image name within the data item hierarchy. Here is an illustration:
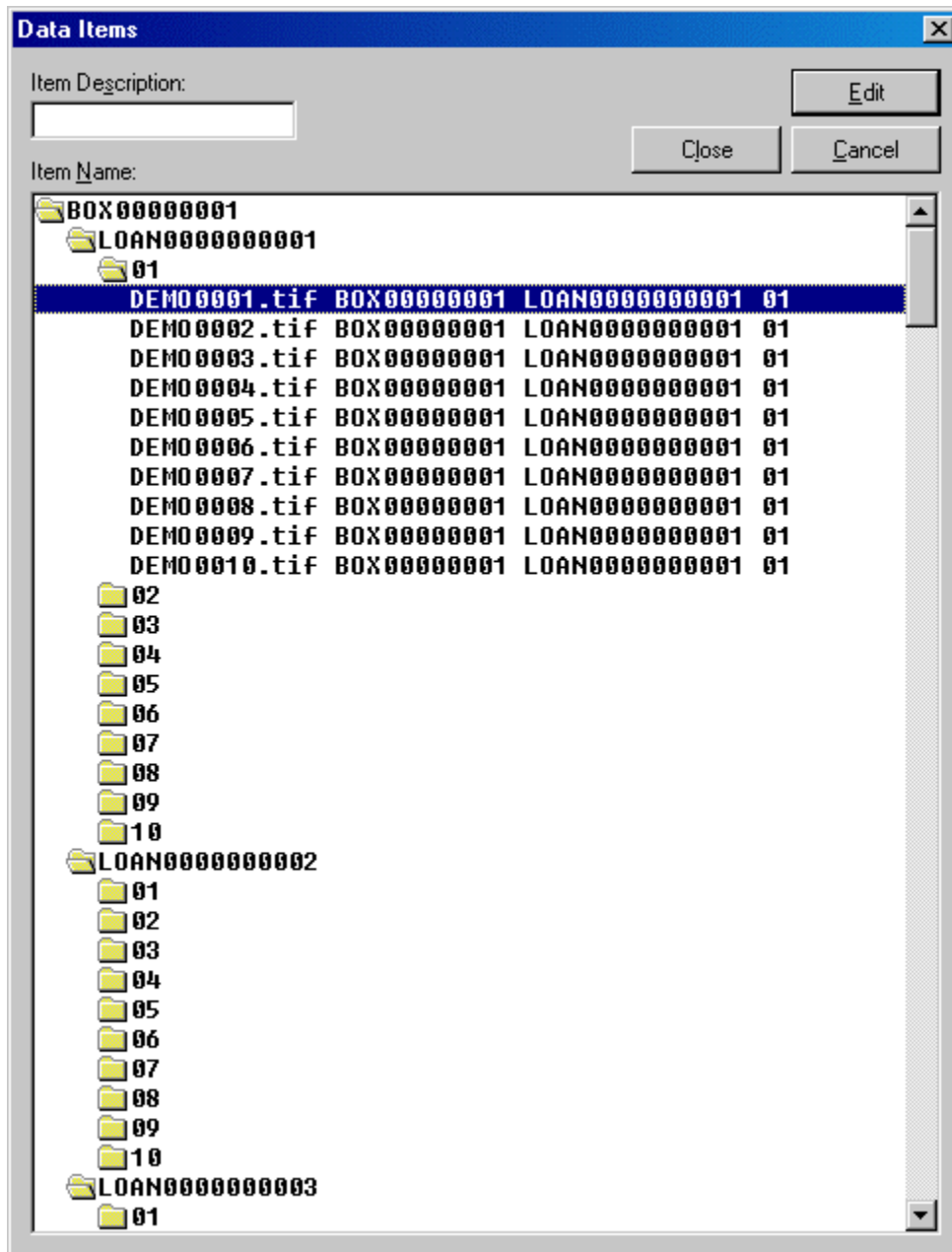
IMAP= Parameter

The IMAP= parameter (Map to INI File) points to the keyboard map INI file, which contains the map to the keyboard key or key combination to display Data Items dialog for the Document Assembly job. See Mapping the Keyboard for more information on this parameter and how it is used. See Mapping a Key to Display the Data Items Dialog for instructions on how to set up a hot key for displaying the Data Items dialog within Document Assembly. For example:

```
IMAP=keybrdmap.ini
```

Indicates that the keybrdmap.ini file, stored in the FormWare\Ini\Keyboard map folder, contains a pointer to keyboard map equivalent (383). This value points to the Data Items dialog for this Document Assembly job.

SCN= Parameter

The SCN=Y parameter (Enable Scan Operator) enables the Scan button on the Image Insert dialog within Completion. Press the Scan button to access the Scan Operator program, where you can rescan an image using the selected image name. The operator can also type the name of the image in the Image Name field. Set this parameter to SCN=N to disable this button. See the BOK= Parameter for an example of the Image Insert dialog.

Designing a Job for Document Assembly

Follow these steps to create a Document Assembly job in Designer:

1. Open Designer and create a new job accepting the default values on the Job Defaults dialog.

2. On the data form, define the appropriate number of Any Type fields to contain the level information coming from the APD parameter in the BDF file. The number of levels defined in your application (identified by the FWLEVn= parameter in the BDF) determines how many fields you need to create here. Create the fields in the same order that level information displays in the APD= parameter from the BDF file. For example for a value of APD=BOX00000001 LOAN0000000001 01 1, create the following fields:
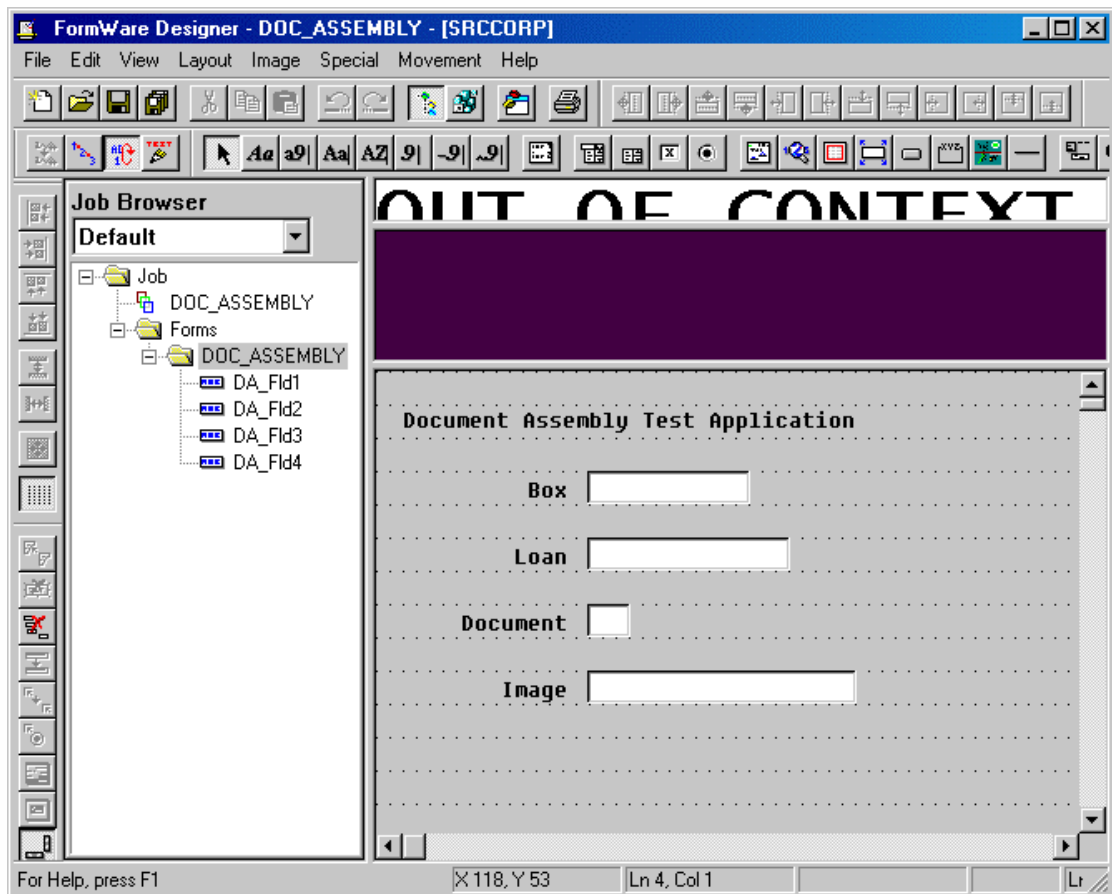
Note: Because the APD data is inserted on the form exactly as specified, make sure that the size of each field is exactly large enough to contain the appropriate level data. If you have a character space between the level values in the APD parameter (e.g., APD=BOX00000001 LOAN0000000001 01 1) be sure to design the fields to accommodate that extra space or values inserted into the group of fields will be off by one.

3. Select **Save All** to save your work.

4. Minimize the reject repair ticker tape and the image viewport areas on the Designer window by grabbing the windowpane handles and dragging upward. This will maximize the data form area displayed in Completion, which is the only area used in a Document Assembly application. For example:
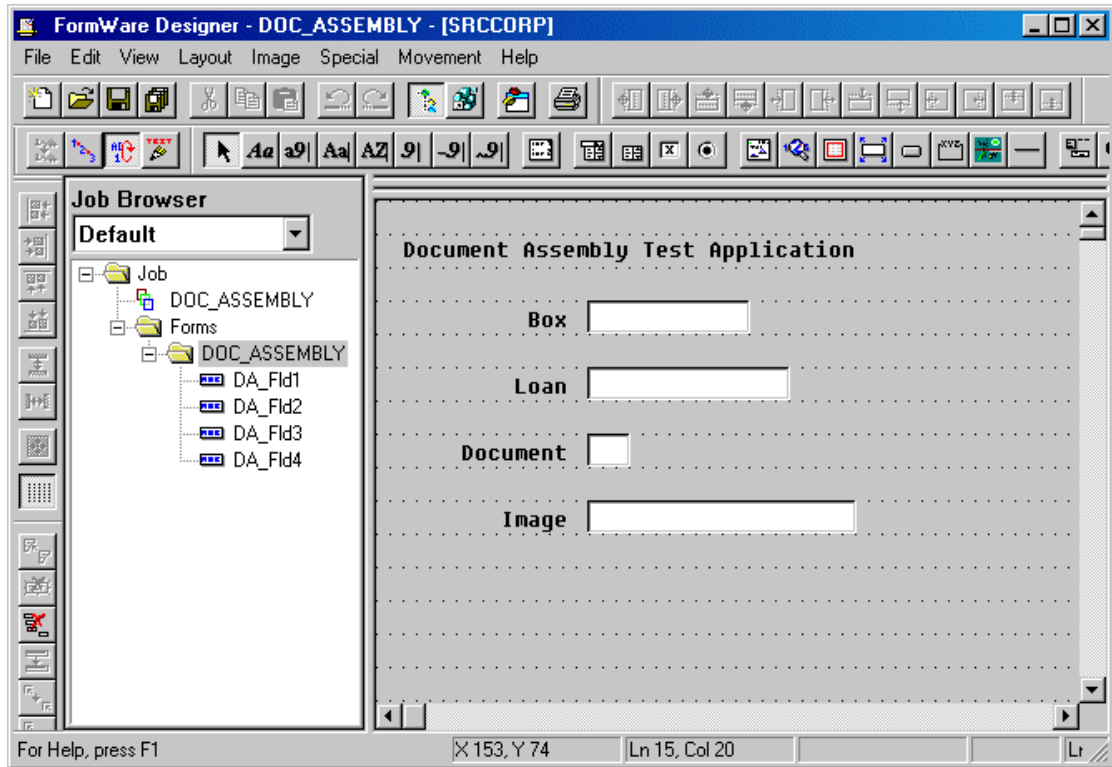
Note: If you have more than one form within your job, make sure that you minimize these panes on all of the forms within the job or the settings will not take affect.

5. Display the Job Properties, Image tab and set properties for the attachment viewer. Document Assembly uses the attachment viewer properties to display the images in Completion. The following attachment viewer properties are appropriate for Document Assembly:
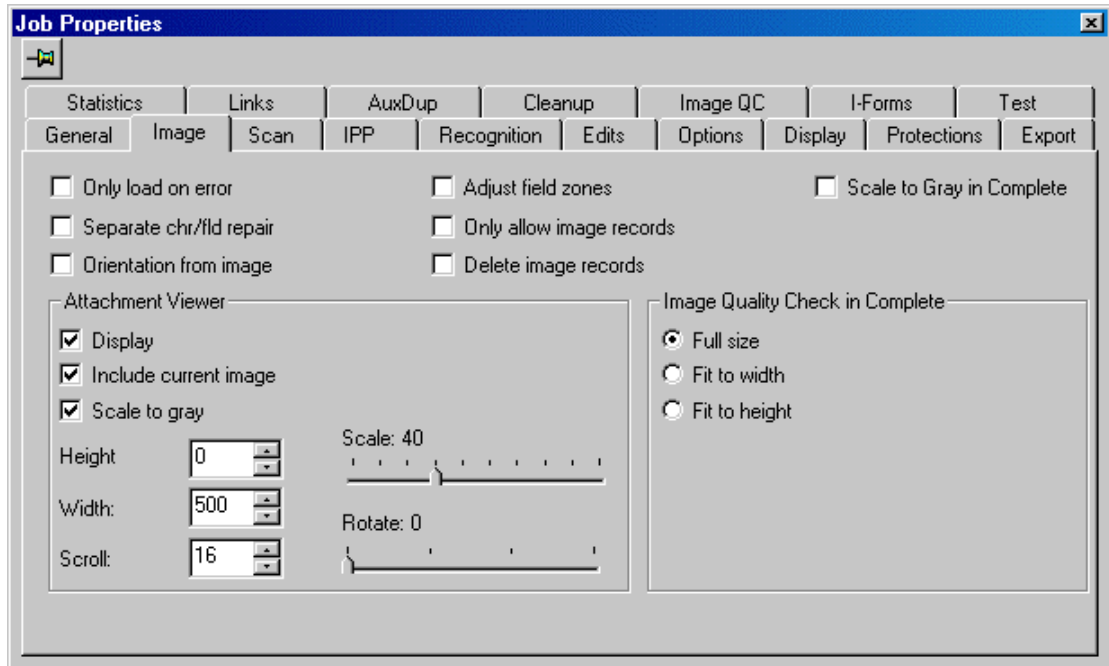
6. Display **Edit > System Defaults > Form and Field**. Set the Form Layout to **Horizontal**.

7. Select **Save All** on the Designer window to save your work.

Mapping a Key to Display the Data Items Dialog

The Data Items dialog is an essential part of any Document Assembly application. Completion operators use this dialog to view the data item hierarchy within a collection of documents and perform maintenance tasks on that data if necessary. Here is an illustration of the Data Items dialog:
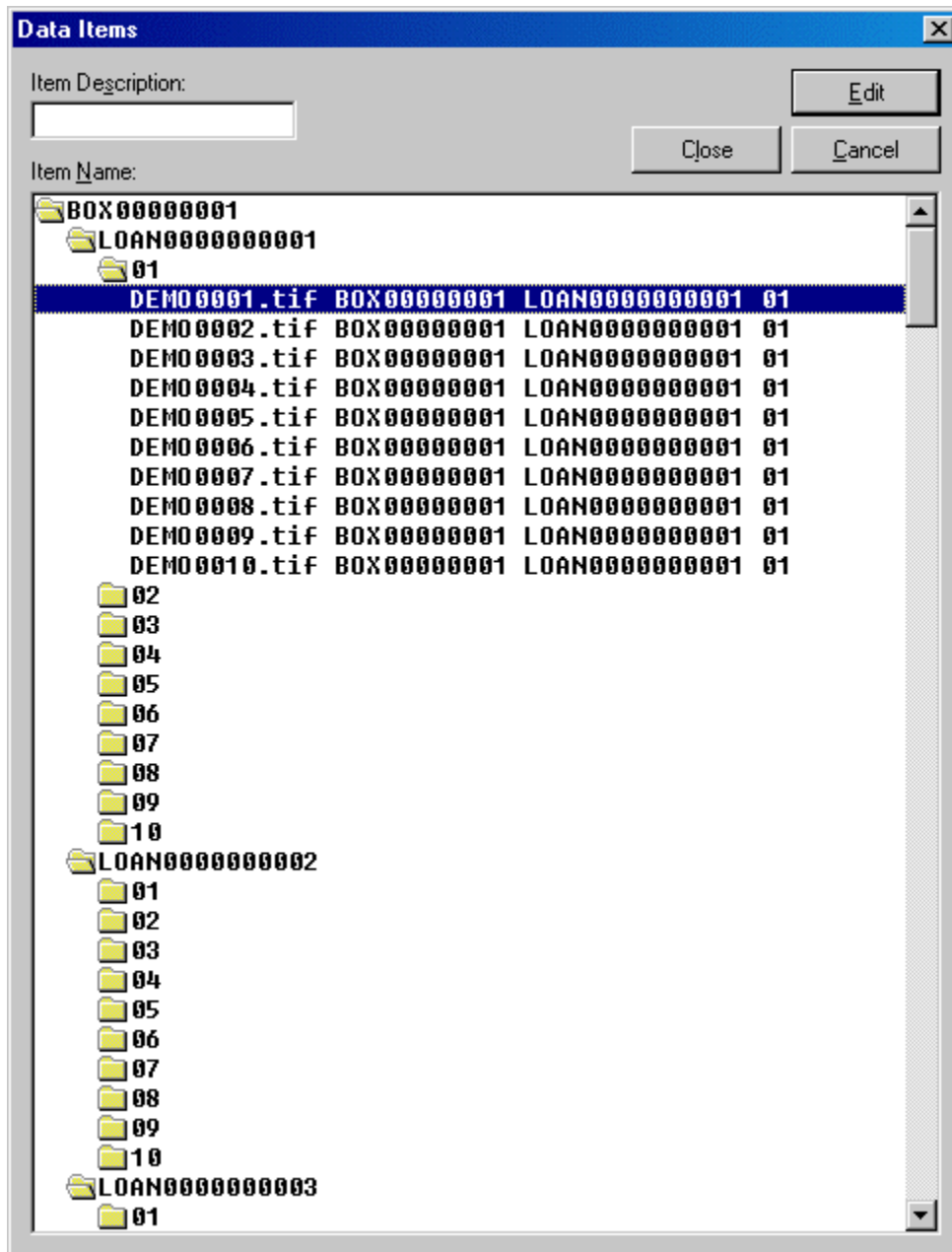
The Data Items dialog is activated using the keyboard equivalent '383'. You can map this function to a key on the keyboard. We recommend that you use the Control (CTRL) key in combination with another key on the keyboard for all keyboard maps within this mode of Completion, since the Document Assembly functions are built to recognize a CTRL key mapping convention while the Data Item Dialog Viewer is active. When the initial view of Document Assembly requires the Data Items dialog to be active, use the VBA PushKey function to display the dialog during an OnTransactionStart event.

By design, when the Data Items dialog displays, it will lay over top of the data form portion of the Completion window. The Data Item dialog displays a tree view of the data hierarchy passed to Completion for processing. The hierarchy is built based on the level tags (FWLEVn) within the BDF file.

Note: Data item level tags are programmatically inserted into the BDF within the FormWare Database System. If you are using the FormWare File System, use the SetDataProperty within a VBA edit task prior to Document Assembly to insert these tags into the BDF file.

The steps in this section show you how to create a job-specific keyboard map INI file, which maps the Data Items dialog to a particular keyboard key within Completion. You will then force display of the Data Items dialog using the VBA PushKey function within an OnTransactionStart event.

1. Open Administration.

2. From the File Browser, display the \Ini\Keyboard maps folder.

3. Right-click in the Contents area and select **New** from the context menu. This displays a text editor.

4. Choose the key or key combination you want to use for your hot key and then use the appropriate three-digit ASCII value for that key. See Mapping the Keyboard for more information. Combine this keyboard map value with the ASCII value for the Data Items dialog, which is "383". For example:

   ```
   C097=383
   ```

5. Displays the Data Items dialog in Completion when you press the [CTRL]+[a] key combination.

6. Select the **Save** button to save your changes to the new INI file.

7. In the Save As dialog, type a name for this INI file and select **Save**.

Note: You can choose to store each keyboard map as a separate INI file (organized by job) or save all keyboard maps in the same file under a more generic name. If you save them in the same file, you should add comment lines within the INI file to identify the purpose of each keyboard map. Insert a # sign at the front of each comment line to designate it as a comment.

8. Select **File > Exit** to return to Administration.

9. Open your Document Assembly job within Designer.

10. Select the **VBA** button on the toolbar to open the VBA Editor for this job.

11. Double-click on ThisDocument in the Project Browser.

12. Define an OnTransactionStart event for the hot key you just defined.
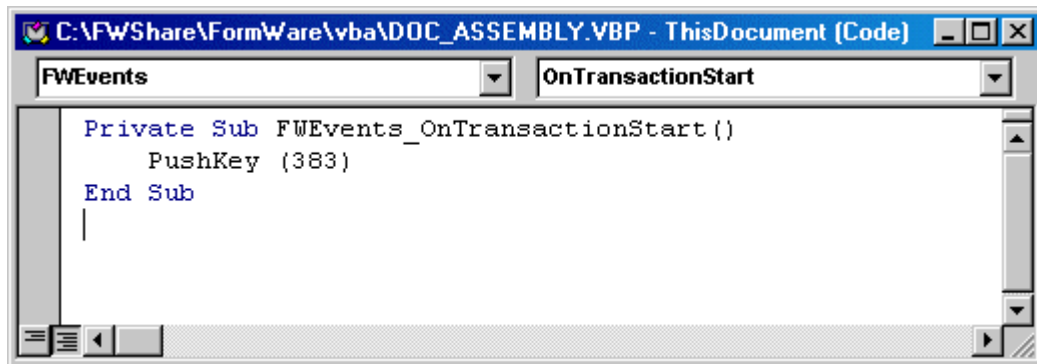
13. Select **File > Save** to save your changes.

14. Select **File > Close** to return to Designer.

15. Select **Save All** on the Designer window to save your changes.

Testing the Document Assembly Job

Before you move any job to production, you should run all processes within a test environment first to make sure they work. This includes running your job against a group of test images. To do this, collect a group of test images and place them in the FormWare\Images folder. Then create a test BDF file which points to these test images. For Document Assembly jobs, it is also important to test the BDF parameters that are used in Document Assembly, such as the FWLEV and ADP parameters.

Follow these steps to test your Document Assembly job:

1. Import or scan a collection of test images for your Document Assembly job. Place these images in the \FormWare\Images folder.

2. Create a test BDF for Document Assembly in the \FormWare\Files folder. See Understanding Document Assembly BDF Parameters and use the BDF illustration there as a guide. See Working with Batches for instructions on how to create a BDF file for testing purposes.

3. Create a job INI file with the appropriate parameters for the Document Assembly job in the \FormWare\INI\Job folder. See Understanding Document Assembly Job INI Parameters and use the INI file illustration there as a guide. See Creating a Job INI File in the Setting INI Parameters section of Designing.

4. Create a Document Assembly job within Designer. See Designing a Job for Document Assembly for instructions.

5. Map a key or key combination to display the Data Items dialog within Completion. See Mapping a Key to Display the Data Items Dialog for instructions.

6. On a command line, type the following to test the behavior of a job that performs only Document Assembly functions on a batch:

```
complete –BLD=BD c:\fwshare\formware\files\<filename>.bdf
```

Where <filename> is the name of the test BDF file you created.

Or,

Type the following to test the behavior of a job that performs both OCR recognition and Document Assembly functions on a batch:

```
complete –BLD=YD c:\fwshare\formware\files\<filename>.bdf
```

7. View the batch in Completion. The Data Items dialog should display over the data form portion of the screen. Select **Cancel** to close the Data Items dialog.

8. Test the key (or key combination) you mapped to the Data Items dialog to see if it redisplays the dialog.

9. Test the functionality of the Data Items dialog. See Running a Document Assembly Application in Using for instructions.

10. **Close** the batch and **Exit** Completion.

Creating a Document Assembly JobFlow

Once command line tests of the job are successful, you can create a Document Assembly JobFlow. For applications that are performing only Document Assembly functions, the JobFlow you create will have only one step in it.

Follow these steps to create a Document Assembly JobFlow:

1. From Administration, select the **JobFlow Designer** button to open JobFlow Designer.

2. Select **New** to create a new JobFlow. The New JobFlow dialog displays.

3. In the **New JobFlow Name** field, type the name of this JobFlow and then select **OK**.

4. Select the **Completion** button on the JobFlow Task toolbar.

5. On the Completion Modes dialog, select **New** as the Processing Mode and then **OK**.

6. Double-click on the **Completion – New** task in the JobFlow Browser to display the Task Properties dialog.

7. In the **Client Parameters** field, type:

   ```
   -e –BLD=B –VBP=<VBA project name>.vbp
   ```

8. Where <VBA project name> is the name of the job VBP file where you have defined the PushKey function to display the Data Items dialog. VBA project files reside in the \FormWare\VBA folder. For example:
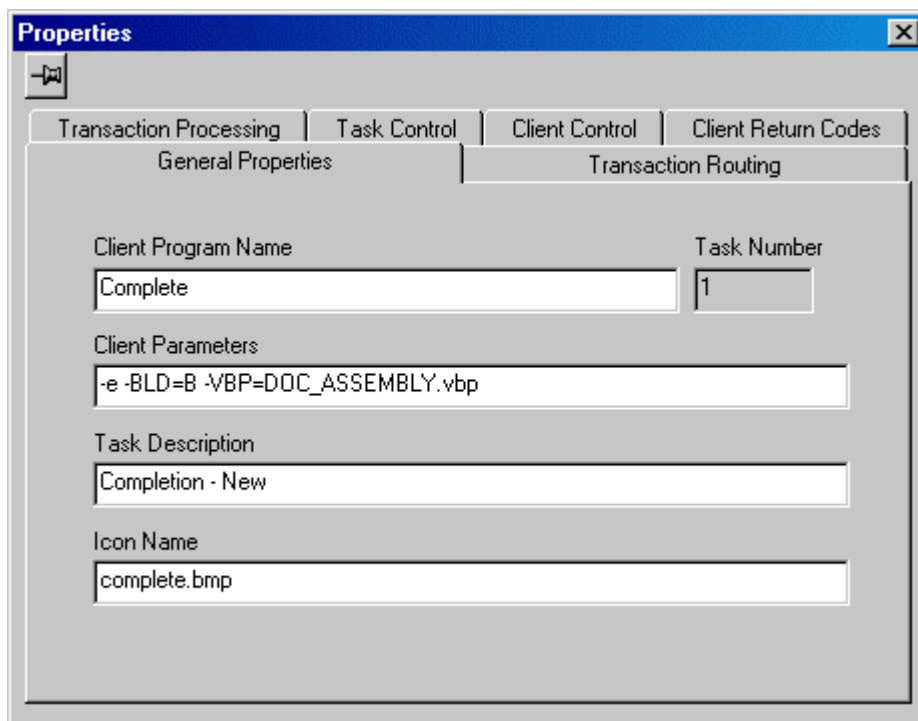
9. **Close** the Task Properties dialog.

10. **Save** your work and **Exit** JobFlow Designer.

Mapping the Keyboard

The keyboard INI files contain the keyboard layout to be used during a Completion session. Each keyboard map has an .ini extension. To apply a keyboard map, you must specify the IMAP= parameter in an INI file that points to the keyboard INI file with the desired keyboard configuration.

Within your keyboard map INI file you can use the following prefixes:

- K - Numeric value of any key on the keyboard (e.g., K002 is the [F2] key).
- SF - Combination of the [SHIFT] key plus a function key (e.g., SF02 is [SHIFT]+[F2]).
- CF - Combination of the [CTRL] key plus a function key (e.g., CF02 is [CTRL]+[F2]).
- AF - Combination of the [ALT] key plus a function key (e.g., AF02 is [ALT]+[F2]).

A sample keyboard map INI file contains the following values:

```
K002=016
SF02=283
CF02=294
AF02=278
```

The number to the right of the equal sign represents the key on the keyboard. These key numbers are listed in the table below. For example, the key value 283 is the Field Correct key. During Completion the operator enables the field-correct condition by pressing [Shift]+[F2].

See the Setting INI Parameters section for more information on setting INI parameters within FormWare.

Creating a Keyboard Map File

To create a keyboard map:

1. Open Administration.

2. From the File Browser, display the \Ini\Keyboard maps folder.

3. Right-click in the Contents area and select **New** from the context menu. This displays a text editor.

4. Type the appropriate keyboard map parameters. For example:

5. Select the **Save** button to save your changes to the new INI file.

6. In the Save As dialog, type a name for your INI file and select **Save**.

7. Exit the text editor and return to the Administration window.

Using Keyboard Map Values

The following table lists the values for the physical keyboard keys from 002 to 029

Number	Corresponding Key	Number	Corresponding Key
002	F2	015	F8
003	F3	016	F9
004	Cursor Down	018	Cursor Right
005	F4	021	Cursor Up
006	F5	022	Insert
008	Backspace	023	Delete
009	Tab Key	024	Page Up
011	F6	025	Home

Number	Corresponding Key		Number	Corresponding Key
012	Cursor Left		027	Escape
013	Enter		028	End
014	F7		029	Page Down

The following table lists the values for the keyboard functions from 257 to 490.

Number	Corresponding Key		Number	Corresponding Key
032-126	Corresponding ASCII values		328	Field Edit Key 4
257	Field Release		329	Field Edit Key 5
258	Auto		330	Field Edit Key 6
262	Dup		331	Field Edit Key 7
263	Current Record		332	Field Edit Key 8
264	Oversign		334	Document Back
266	Release		335	Document Forward
267	Insert		336	First Document
268	Delete		337	Last Document
269	Record Back		338	Specify Document
270	Aux Dup		339	Insert Document
271	Record Dup		340	Delete Document
272	Record Forward		345	Image Back
273	Cursor Up		346	Image Forward
274	Cursor Left		347	First Image
275	Cursor Right		348	Last Image
276	Cursor Down		349	Specify Image
277	Tab Key		350	Zoom Up
278	Form Change		351	Zoom Down
279	Job Help		352	Reset Image
280	Form Dup		353	Display Full Image
281	Field Back		354	Rotate Image 90
282	Field Forward		355	Rotate Image 270
283	Field Correct		356	Rotate Image 180
284	Field Delete		357	Invert Image
285	Not Longer Used		358	Load New Image
286	Specified Field		361	Insert After
287	Field Write		362	Process PUSHKEY values after performing field edit.
288	Field Flag		363	Scroll Image Up
291	No Longer Used		364	Scroll Image Down
294	Record Correct		365	Scroll Image Left
295	Form Help		366	Scroll Image Right
296	Record Insert		367	Display Field List
297	Record Delete		371	View Attachments

Number	Corresponding Key		Number	Corresponding Key
299	Reject menu		375	Record Accept
307	Form Dup Out		377	Table Release
308	Field Override		379	Clear a Rejected Record
309	Field Edit Key 1		465	Close Batch
311	Previous Reject		466	Interrupt Batch
314	Word Delete		470	Display Accumulators
315	End of Field		471	First Record
316	Word Left		472	Last Record
317	Word Right		483	Suspend Session
319	Non-destructive Enter		487	Record Write
320	Verify Override		489	No Longer Used
325	Character Flag		490	Preview Stats
326	Field Edit Key 2		519	Toggle Original/Enhanced Image
327	Field Edit Key 3		523	Display Ruler

Defining Export Settings

Exporting data from FormWare is one of the final steps in the data capture process. You use export programs to format data from a batch or group of batches to prepare it for printing or for export to a text file, PDF, or some other backend repository. Using the Administration module, you can also conduct an FPL export process on a batch or a group of batches ad hoc.

Output data from several export processes, such as Unisys InfoImage RBE Export, Optika eMedia Export, and XML Export, is stored in the Results folder. The first time one of these export options is used, it creates a default directory with the name of the export process (e.g., RbeClnt, OptClnt, and XMLClnt). The output files created by each export process are then stored in the appropriate directory. Alternatively, when you configure your export process, you can designate a different output directory for the export files.

Creating an Export Program

To create an export program:

1. Open Administration.

2. From the File Browser, select the Export folder.

3. Right-click in the Contents area and select **New** from the context menu. This launches a text editor.

4. Type the source code for your program.

5. Select **Save** in the text editor to save your program.

6. Close the text editor and return to Administration.

Modifying an Export Program

To modify an existing export program:

1. Open Administration.

2. From the File Browser, select the Export folder.

3. In the Contents area, double-click on the .et file you want to modify. This launches the text editor and displays the source code.

4. Edit and save the source code as necessary within the text editor.

5. Exit the text editor and return to Administration.

Parsing an Export Program

To parse an export program:

1. Open Administration.

2. From the File Browser, select the Export folder.

3. In the Contents area, right-click on an export program file with an .et extension.

4. Select **Parse** from the context menu to compile the program and create the .R file. To re-parse an export file, double-click on an existing .R file in the Export folder.

Understanding the Export Options

FormWare offers the following export options. The following sections describe how to set up these export processes:

FileNET	Using the FileNET export component you can export data directly from FormWare into a FileNET repository. Whereas FormWare has historically conducted forms processing, FileNET produces image management processing systems, referred to as document processing. In document processing, the emphasis is on capturing just enough information off an image to identify the image and then store the image with that information. In forms processing, the emphasis is on capturing as much information off the image as possible and then the image is discarded.
IMR Alchemy	The FormWare IMR Alchemy export process transfers images and/or index information from FormWare to the IMR Alchemy document database for storage. Once submitted through the IMR Alchemy export component, images are available for query and retrieval by end-users of the IMR Alchemy system.
ODBC	When using ODBC, you export to a data source. Example data sources are: dBase database, Excel spreadsheet, FoxPro database, MS Access database, Oracle database, or a text file.
Optika eMedia	Using the Optika eMedia™ export component you can export index data and images into the Optika eMedia document management system. You can later use these documents for archiving and retrieval. Electronic report management, transaction viewing, and workflow processing are all facilitated through the Optika eMedia document management system.

OTG	FormWare tightly integrates with Online Technologies Group, Inc. (OTG) Application Xtender (AX) system resulting in a simple and effective document capture, storage and retrieval product. FormWare provides document scanning, image and data quality assurance, and data capture. The integrated product allows you to easily access stored documents in the OTG AX document management system.
Unisys InfoImage (RBE)	Use the Unisys InfoImage Rapid Batch Entry (RBE) export component to export index data and images directly from FormWare into a Unisys InfoImage workflow system. You can use these images for archival, entry into the InfoImage workflow, or both. Once submitted into InfoImage, the images are available for query, retrieval, printing, and routing by end users of the InfoImage system.
XML	The XML export component allows you to export data to an XML-formatted, flat text file. The export component will generate one output file per transaction or batch. The filename is the same as the BDF filename, but with an .XML extension and resides in the same directory as the BDF.

Setting Export Job Properties

You must set properties on the Protections and Export tabs on the Job Properties dialog to establish an export process. The properties on the Protections tab prohibit exporting batch data, based on certain criteria. These settings must be satisfied before you can export a batch. The export protections are applied to one batch at a time, even if the export is configured to export a group of batches all at once.

An export protection is applied to one batch at a time, even if the batch is grouped using the List Options in the export device. The export protections are not applied to a group of batches as a whole. It is determined whether a batch should be exported and then the protections are applied to the next batch in the group.

The properties on the Export tab impact how data is exported. These properties only apply to exporting data using the default programs or custom FPL programs.

To set these job-level properties:

1. Open a job in Designer.

2. Display the Job Properties, Protections tab for the job.

    Each export protection type on this tab has three states: OFF, NEGATIVE, and POSITIVE. The **None** option for each of the export protections is the OFF state. This OFF state is followed first by the NEGATIVE state and then by the POSITIVE state for the export protection. For example, select **No Error Flags** to export only batches that do *not* have any error flags or select **Error Flags** to export only batches that *do* have error flags. The **No Error Flags** option is the negative state of the export protection.

3. In the **Terminate** area, select the option that verifies whether the batch has been closed instead of interrupted during the Completion session:

    o **None** - Protection is ignored.

    o **Not Terminated** - Exports the batch if it is not terminated from the Completion module.

    o **Terminated** - Exports the batch if it has a status of terminated (rather than interrupted). When a batch is interrupted in a Completion session, it is not considered closed.

4. In the **Export** area, select the option that verifies whether the batch has been previously exported.

   o **None** - Protection is ignored.

   o **Not Exported** - Exports the batch if it has not been previously exported.

   o **Exported** - Exports the batch if it has been previously exported.

Note: To use Export protection, you need to select the Edit button for the Export Device on the Job Properties, Export tab and select the **Mark as Output** check box on the Properties dialog. When you select Mark as Output, the information in the batch header is modified when the batch data file is exported. Export protection then searches for this information in the batch header to determine whether a batch has been previously exported.

5. In the **Error Flags** area, select the option that indicates whether error flags are added to a batch during a Completion session. Operators can then remove flags in the Completion - Remove Flags mode.

   o **None** - Protection is ignored.

   o **No Error Flags** - Exports the batch if it does not contain error flags.

   o **Error Flags** - Exports the batch if it contains error flags.

6. In the **Batch Log** area, select the option that indicates whether batch log files should impact the batch export process. A batch log file is a text file that contains error or audit log information. Batch log files are stored in the \FormWare\Files folder. Output statements from an FPL Field Edit or File End Edit program generate batch logs. You can identify batch log files by their .log extension.

   o **None** - Protection is ignored.

   o **No Batch Log** - Exports the batch if it does not have a batch log.

   o **Batch Log** - Exports the batch if it has a batch log.

7. In the **Verified** area, select the option that indicates whether a batch must pass the Verify mode of Completion before it can be exported.

   o **None** - Protection is ignored.

   o **Not Verified** - Exports the batch if it has not been verified.

   o **Verified** - Exports the batch only if it has been verified.

8. In the **In Balance** area, select the option that indicates whether accumulators within a batch must be in balance (i.e., equal to 0) before the batch can be exported. An accumulator is one of 16 numeric data storage areas within a data form. You can add field data to or subtract it from the accumulator value. You set accumulator options on the Field Properties, Selective Edits tab in Designer.

   o **None** - Protection is ignored.

   o **Not In Balance** - Exports the batch when accumulators within the batch are not in balance.

   o **In Balance** - Exports the batch only when all accumulators within the batch are in balance (i.e., all equal to 0).

9. An error message displays to the operator when a violation against an export protection is encountered during an FPL export session. The Export Protection Violation message displays the name of the failed batch, the export protection(s).

10. Display the Job Properties, Export tab for the job.

11. To ensure that all of the records you are exporting are the same size, set the following fixed records values. When you are exporting batch data you may need to produce records that contain the same number of characters, regardless of the number of characters that make up the data. This may be in the case of a mainframe that cannot handle records of variable lengths or for storing data on a magnetic tape.

   a. In the **Record size** field, select the total number of characters in the largest record that the export program can generate. The default record size is 4,096 and the maximum record size is 32,767.

   b. Select the **Fixed length records** check box to forces all exported records to the same length. When you select this option, the export program inserts pad characters in records to make them all the record size specified in the **Record size** field. Fixed length records are used mainly for tape devices and are measured by the number of characters. The maximum value is 32,767

   c. In the **Pad character** field, type the character to be added to records to make them a fixed length. The default pad character is a space. An operator can type another character to distinguish the pad character from the actual record data.

12. If the exported data are to be blocked, set the following field values. Records are commonly output in blocks to save time when writing data to a magnetic tape.

   a. In the **Block size** field, select the number that determines the block size. This number must be an exact multiple of the record size. The default block size is 4,096 and the maximum block size is 32,767.

   b. Select the **Blocked records** check box to output data files with the specified block size. Records are commonly output in blocks to save time when writing data to a magnetic tape.

13. In the **Defaults** area, select from the following values to determine the format of the exported data:

   o **ASCII Format** - Exports data in the ASCII character set. Record data is exported in a stream of ASCII characters, with no delimitation between fields. Each stream of record data is followed by a line feed (LN) and carriage return (CR) so that record data is on a single line of text. All record data is appended into one line. The following is a sample data file exported with the ASCII export format:

   ```
   JOHN BOLAND   453219876 891 SCENIC HIGHWAY   NEW ORLEANS   LA70115
   ```

   o **Comma Delimited** - Exports data in a comma delimited format, separating the data from each field with a comma. The following is the syntax: "<data>","<data>","<data>","<data>". This is a common export format for exporting to databases. Each line in the text file contains all of the data for a record. The following is a sample data file exported with the Comma Delimited export format:

   ```
   "MARY HOGAN","434534534","2519 CREEK DRIVE","DIXON","MA","36840"
   ```

   o **Form Defined** - Exports field data according to the export positions set in Designer on a field-by-field basis. No other formatting is applied to the export data. The export positions are specified on the Field Properties, Checkbox Edits tab. A device is not optional, it must be specified for this option.

14. Select the **Browse** button next to the **Export program** field, to select an FPL export program to be used with the job. Export programs are located in the \FormWare\Output folder. Select the adjacent **Edit** button to view the export program in the text editor. You can then modify the syntax of the export program. Select the **Compile** button to parse the program file in the Export Program field and check for certain syntax errors. Parsing a program converts the

syntax of its source code to a binary file, which is then used in processing. This binary file has the same name as the corresponding program file, except with an .r extension.

Note: See the following sections for instructions on selecting an export program and export device depending upon the type of export you want to perform.

15. Select the **Browse** button next to the **Export device** field, to select the export device to receive the batch information when exporting. This device must be in the \FormWare\Devices folder before it is attached to a job. You must specify the device name to execute a sort or stand-alone file edit.

16. Select the **Edit** button to display the Properties dialog for the specified device. The properties you specify for the device are applied to batches exported using the current job properties.

17. Select **Save All** on the Designer window toolbar to save your changes.

Configuring an ODBC Export

Data captured with FormWare can be exported to multiple locations, including a database. The database technology that is compatible with FormWare is Open Database Connectivity (ODBC). When you export to a database, use the ODBC Export JobFlow task. You can export data records in a batch data file to a single database table, multiple tables in a single database, or multiple databases. FormWare exports data one field at a time, based on the keying order of the data fields.

When using ODBC, you export to a data source. Example data sources are a dBase database, an Excel spread sheet, a FoxPro database, an MS Access database, an Oracle database, or a text file. The discussion concerning exporting FormWare batch data to an ODBC data source focuses on exporting to a database, although other sources are possible.

Note: You must install the ODBC drivers on all workstations that will use FormWare to export to an ODBC database. Refer to your ODBC provider for installation instructions. If you attempt to run an ODBC export process on a workstation without the ODBC drivers, you will receive an error.

Before you configure the FormWare data fields for export to an ODBC data source, finalize the design of both the FormWare job and the corresponding ODBC data source. This may take some planning time to ensure that the fields in both the job and the data source conform to the same data type, length, and any other considerations. You need to be cautious when assigning properties to the data source fields, such as not allowing duplicate information. To do so can impede the export process. FormWare does not check during the design phase to verify that the ODBC data source properties are compatible with those in the FormWare job. Conflicts may only become apparent during export.

Caution: The ODBC database table and field names can only contain A-Z, a-z, 0-9, and the underscore character (_). If you include an invalid character, such as a space, you will experience problems.

Once you construct a corresponding ODBC data source, you must map the FormWare data fields to the database fields. Access the ODBC configuration properties for FormWare within the Designer module at either the form or field levels. The General pages on both Form and Field Properties dialogs in the Designer module contain a Component field with a drop-down list of export components, including ODBC Export. Once you have selected the ODBC Export option, you can then select the Configure button to access the Form and Field Property dialogs.

Setting ODBC Form and Field Properties

You can configure ODBC properties for a field from either the Field Properties dialog or the ODBC Form Properties dialog. The following steps explain how to configure FormWare data fields for export to an ODBC data source from the ODBC Form Properties dialog.

1. Open a job and form in Designer.

2. Display the Form Properties, General tab for the form.

3. In the **Components** field, select the **ODBC Export** option.

4. Select the **Configure** button to display the ODBC Form Properties dialog.

5. Select the **Browse** button next to the **Default Data Source** field to display the Select Data Source dialog where you select the name of the data source used as the default value when a new field is configured.

6. Select a data source configured for the current data form. If you cannot access the data source from the current workstation, type the data source name in the **DSN Name** field. Select **OK**.

7. Select the **Browse** button next to the **Default Table** field to display the Select Table dialog where you select the name of the data source table used as the default value when a new field is configured. The **Browse** button for the Default Table field is not enabled until you select a default data source.

8. Double-click on a table name in the list to select it. If you cannot access tables from the current workstation, type the name of the table into the field.

9. Select the **Add** button. The Select Column dialog displays.

Note: The data source field is also referred to as a column.

10. Double-click on a field name to select it. This displays the ODBC Field Properties dialog with the selected default database, default table, and field.

11. Set any additional field options and select the **OK** button. The name of the FormWare field displays in the ODBC Form Properties dialog with its corresponding database, table, and field.

12. Select the **Start of Record** check box to create a new database record for each record in the batch data file during the ODBC export process. A new database record is created in each of the database tables specified for the FormWare ODBC-configured fields.

    For example, if you select this option and all the ODBC-configured fields on the data form are specified to database Table1, then one new record is created in Table1 for each record in the export from the batch data file. If three of the FormWare fields are specified to database Table1 and one field is specified to database Table3, then a new record is created in bothTable1 and Table3 for each record exported from the batch data file. This option can also be set for individual ODBC-configured fields. See the Understanding the Start of Record Option section for more information.

13. Select the **Add** button to configure a FormWare field to export to an ODBC data source. (You can also select the **Configure** button on the Field Properties, General tab for the **Component** field.) The Select Field dialog displays.

14. Double-click on a field name. The ODBC Field Properties dialog displays for that field.

15. In the **Data Source Name** field, select the **Browse** button to select the data source for this field. The data in the FormWare field is exported to this data source during an ODBC export

process. If you cannot access the data source from the current workstation, type the data source name in the field.

16. In the **Table Name** field, select the **Browse** button to select the database table for this field from the list of tables available for this data source. The data in this field is exported to the selected table during an ODBC export process. The **Browse** button is not enabled until you select a default data source. If you cannot access the database table from the current workstation, type the name of the table in the field.

17. In the **Column Name** field, select the **Browse** button to select the database field for the current FormWare field. The data in the FormWare field is exported to this database field, or column, during an ODBC export process. The **Browse** button is not enabled until you specify a database table. Double-click on a column name to select it and return to the ODBC Field Properties dialog. If you cannot access the database column from the current workstation, type the name of the column in the field.

18. Select the **Start of Record** check box to create a new database record in the specified database table during the ODBC export process. Export data continues to be added to the database record until either another Start of Record is encountered or until the end of the data in the batch data file. **See the Understanding the Start of Record Option section for more information.**

19. Select the **Trim Left** check box to remove any white space from the left of the field data to clean up the exported data. This includes either space or tab characters. For example, if the FormWare field contains "_ _ _ XYZ" where "_" is a space character, this option removes the spaces and returns only the "XYZ" data.

20. Select the **Trim Right** check box to removes any white space from the right of the field data to clean up the exported data.

21. Select the **No Quotes** check box to omit quotation marks from the field data.

22. Select **OK** to return to the ODBC Form Properties dialog, which displays the following field information:
    o **Field Name** - Displays the name of the ODBC-configured FormWare data field on the current data form. Fields designated as Form Identification, Registration, or Table Zone fields within FormWare are not available for configuration for ODBC export.
    o **DSN** - Displays the name of the data source name to which the FormWare field is configured for export.
    o **Table** - Displays the name of the database table to which the FormWare field is configured for export.
    o **Column** - Displays the name of the table column, or database field, to which the FormWare field is configured to export data.
    o **SOR** - Displays an X when the corresponding FormWare field is designated as a Start of Record on the field level.

23. Highlight a field in the table and select the **Edit** button to modify a FormWare field for export to an ODBC data source. The ODBC Field Properties dialog displays.

24. Highlight a field in the table and select the **Remove** button to remove the ODBC properties for that field. This does not delete the FormWare field, only its ODBC configurations.

25. Select **OK** to return to the Form Properties dialog.

26. Select **Save All** on the Designer window toolbar save your settings.

Configuring Multiple ODBC Fields

Follow these steps to modify more than one ODBC-configured FormWare field at a time:

1. Open a job and form in Designer.

2. Display the Field Properties, General tab for the form.

3. Select the **ODBC Export** option in the **Component** field.

4. Select the **Configure** button to display the ODBC Form Properties dialog.

5. Highlight a number of different fields in the list. You can also use [Shift]+[Click] to select fields in non-sequential order.

6. Select the **Edit** button to display the ODBC Field Properties dialog. The values on this dialog display when they are the same for each of the selected fields. If the field values are different for the database, table, or column name then no values will display. If the field values are different for the **Start of Record**, **Trim Left**, or **Trim Right** options, a gray check appears in the option checkbox.

7. Set values to apply to all of the selected fields and then select the **OK** button.

Understanding the Start of Record Option

You can set the **Start of Record** option at both the form and field level. Each time the ODBC export process encounters a start of record, it creates a new database record in the table associated with the FormWare field.

You do not need to set the **Start of Record** option when performing a one-to-one export of a single FormWare record to a single database record. The default of the ODBC export is a one-to-one export and does not require any further modification. The **Start of Record** option is used when you want to create multiple database records from one FormWare record.

If you set the **Start of Record** option only on the ODBC Form Properties dialog, the ODBC export process creates a new database record in each of the specified tables for each record in the exported BDF. If you set **Start of Record** option for one of the FormWare fields, the ODBC export process creates a new database record in the table specified for the field when that field is encountered during the export process. Data continues to be exported to the new database record until another start of record is encountered or until the end of the data in the batch data file. FormWare exports data one field at a time in the keying order of the FormWare fields.

Note: The **Start of Record** option is not an end of record function. This means that data continues to be exported into the record until another **Start of Record** option is encountered.

In the following example, the patient and hospital fields are specified for database Table1. Each of the detail fields is specified for Table2 and has the **Start of Record** option set. The keying order for the fields on the form is set from top to bottom.
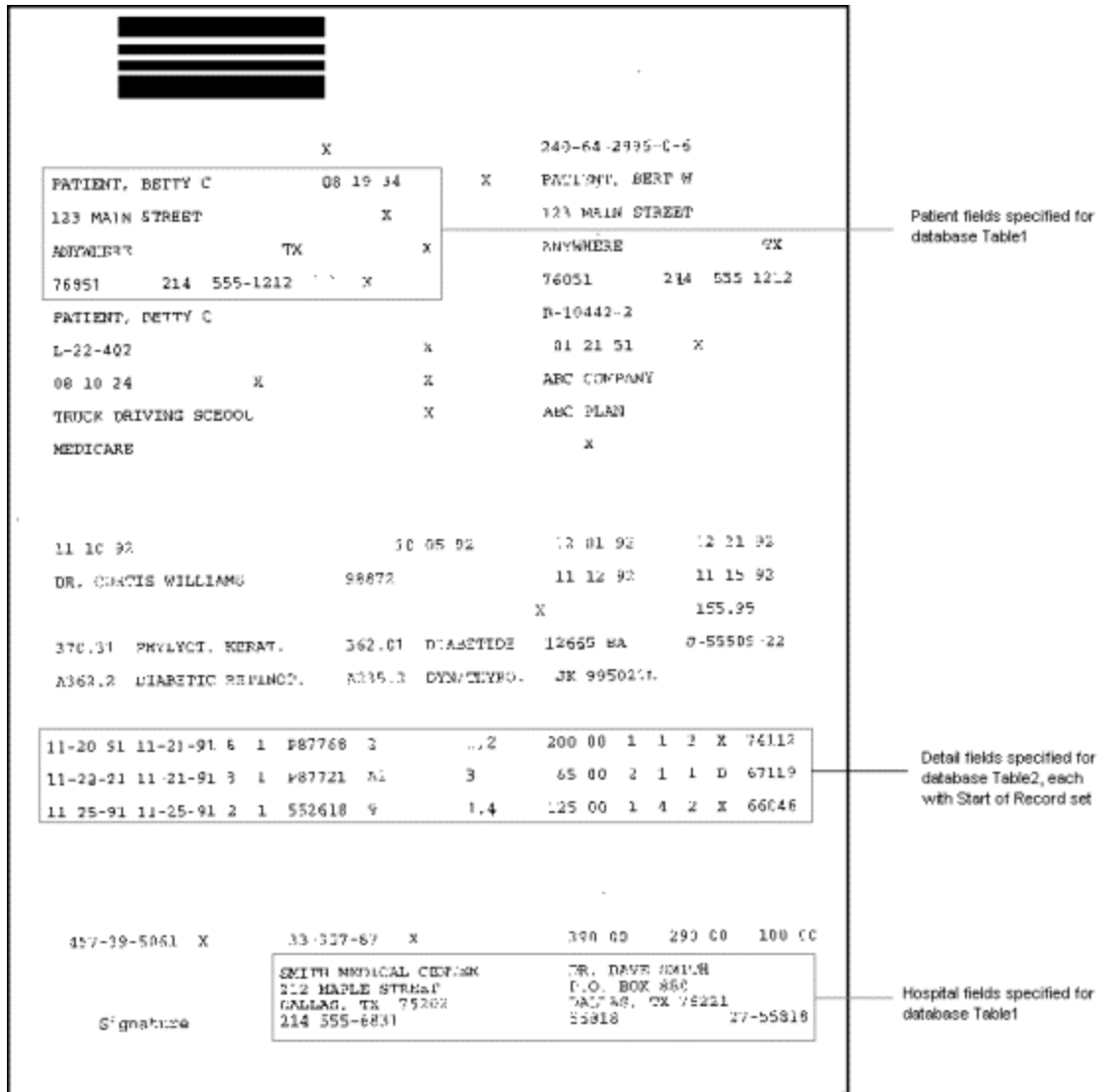
When the first field is exported – a patient name field – a new database record is created in Table1. All of the data contained in the patient fields is exported to that record. When the first detail field is encountered, a new database record is created in Table2 and the data for that field is exported into the new record. When the next detail field is encountered, a new database record is created for Table2 and the data for the second detail field is exported into that new record. This behavior continues for each of the detail fields, since each field has the **Start of Record** option set. When the first hospital field is encountered, that data is exported into the record in Table1 created by the patient fields. All of the data contained in the hospital fields is exported into that same record.

Configuring an IMR Alchemy Export

FormWare tightly integrates with the IMR Alchemy database system resulting in a simple, effective document capture, storage and retrieval product. The FormWare software provides document scanning, image enhancement, image and data quality assurance, and data capture. The integrated product enables users of the combined system to easily access stored documents through Alchemy's ODBC compatible database. Queries, using Windows clients or Web

browsers, can use form fields and/or full-text searches to retrieve any document stored in the Alchemy database.

FormWare IMR Alchemy is the process by which images and/or index information from FormWare is transferred to the IMR Alchemy document database for storage. Once successfully submitted through the IMR Alchemy export component, images are available for query and retrieval by end-users of the IMR Alchemy system.

Understanding IMR Alchemy Output Files

The IMR Alchemy export process uses IMR Alchemy's DataGrab method to automatically import data into a specified database using the associated data file and DataGrabber definition file. Therefore, either Alchemy Build or Search must be installed on your workstation to use the IMR Alchemy export component. The following sections describe the generation of a definition file and data file, and the population of a database during the export process.

Definition Files

The IMR Alchemy export component generates definition files and data files to send to the IMR Alchemy database. The definition (or control) file serves as a table of contents for the different files of a given batch. It specifies the associated images and batch files, as well as the title of each document. The IMR Alchemy export component creates one definition file for each batch.

The form type and page order must be consistent throughout the documents in the batch because the form order in the definition file results from the page order of the first document in the batch. If you scan the pages of subsequent documents in a different order, data will be assigned to the wrong fields.

For example, when the pages of the first document in the batch are scanned in page one, page two, page three order, which becomes the order of the forms in the IMR Alchemy database. If the pages of the second document are subsequently scanned in page two, page one, page three order, problems will occur with data from the second document. Data from page two of the second document will be assigned to the fields on page one in the IMR Alchemy database. Likewise, data from page one of the second document will be assigned to the fields on page two in the IMR Alchemy database.

Every definition file starts with the line and record delimiter statements:

```
Line_Delimiter = ","
Record_Delimiter ("\r\n", Between)
```

Following is an example of a definition file used to find assembly parts in an IMR Alchemy database.

```
Line_delimiter = ","                   // Lines delimiter by a comma
Record_Delimeter ("\r\n", Between)     // Records delimited by "\r\n"
Field ("Document Title", Lines (1))    // Extract the part name
Field ("Part Number", Lines (2))       // Extract the part number
Field ("Assembly", Lines (3))          // Extract the assembly number
Field ("Sub Assembly", Lines (4))      // Extract the sub assembly #
Field ("File", Lines (5))              // Extract the image file path & name
```

Following the line and record delimiter statements are field statements using the output field name as Field-Name, and one statement per output field, listing all the output fields from all the forms in the first document. The file ends with field statements using File or Sub_File as Field-Name, and listing all of the image files in the document. The keyword file is used for the first image, and the Sub_File is used for the subsequent images. Therefore, to access a record's "Part

Name", the user would extract the contents of the "Document Title" line. Similarly, to access a record's "Part Number", the user would extract the contents of the "Part Number" line.

Definition files are stored in the \FormWare\Results folder and are given the batch name plus a .def file extension. Definition files are automatically created and deleted during the export process.

Data Files

The IMR Alchemy export component generates one data file for every batch. All of the output fields and image files from one document are written in one line, separated by commas. Data files are stored in the \FormWare\Results folder and are given the batch name plus the .dat file extension. Data files are automatically created and deleted during the export process.

Populating the Database

The IMR Alchemy export process populates the selected database one batch at a time. After the data file and the definition file are generated, the DataGrab method will be called to automatically import the batch data into the specified database. Before running the export module, make sure that your IMR Alchemy application does not already have the database open.

Setting Up IMR Alchemy Export

Before creating a FormWare job for export to an IMR Alchemy database, you should finalize the design of both the FormWare job and the corresponding IMR Alchemy database. This may take some planning time, to ensure that the fields in both the FormWare job and the IMR Alchemy database conform to the same data type, length, and other considerations. Be cautious when assigning properties to the fields that could impede the export process. FormWare does not check during the design phase to verify that the IMR Alchemy database properties are compatible with those in the FormWare job. Conflicts may only become apparent during export, such as exporting alphabetic data to a numeric-only IMR Alchemy field.

The FormWare job you create must contain fields corresponding to every index field in the IMR Alchemy database to which you want to export data. You can create fields other than index fields on your FormWare forms. These fields can be used to capture data for export to other locations such as a database or text file. Only the fields configured for IMR Alchemy export are referenced in the data and definition files.

After you create a FormWare job, data form, fields, and JobFlow task for an IMR Alchemy export, you must configure the FormWare system for that export. To allow FormWare to export to an IMR Alchemy database, you must select a database at the job level and configure the IMR Alchemy Export component at the form level.

To set up an IMR Alchemy export, you need to set properties at the job, form, and field levels. To do this:

1. Open a job in Designer and display the Job Properties, General tab.

2. In the **Component** field, select the **IMR Alchemy Export** option.

3. Select the **Configure** button to set up a database for all the documents in the job. The IMR Alchemy Database dialog displays.

4. If you have not selected a database for this job, the Title and File name fields will be blank. To change databases or select a new database, select the **Browse** button to display the Open Database dialog.

5. Highlight an existing database and select the **Open** button. The selected database title and file name display in the IMR Alchemy Database dialog.

> **Note**: To ensure successful export to the IMR Alchemy database, the selected database should not have security groups defined. The database should only be secured with a password at the database level.

6. Display the Form Properties, General tab for a form within the job.

7. In the **Component** field, select the **IMR Alchemy Export** option.

8. Select the **Configure** button to display the IMR Alchemy Configuration dialog.

    Fields in the **Output Fields** list will output data to the IMR Alchemy database. The IMR Alchemy export component checks to determine which field names in your job match with field names in the selected database. If your job has not been configured before, by default all the IMR Alchemy matching fields will initially appear in the **Output Fields** list and all fields without matches will initially appear in the **Non-Output Fields** list. To configure your job, you can add fields to or remove fields from the **Output Fields** list.

    The first form in each document must contain at least one output field with a mapping name of **Document Title**. This is necessary because IMR Alchemy uses the **Document Title** field when displaying search results. When a search is performed in an IMR Alchemy database, the **Document Title** field identifies the matching documents. If you do not use a **Document Title** field, IMR Alchemy has no way to report the results of a search.

    Two or more output fields may share the same mapping names. In this case, the field data is defined as the concatenation of all the field data in the keying order.

9. To add a single field to the **Output Fields** list, highlight the field in the **Non-Output Fields** list and select the **Add** button. The IMR Alchemy Field Mapping dialog displays.

10. By default, the **FormWare field name** displays in the **Alchemy field name** list as the mapping name. You can modify this name or select another mapping name.

> **Note**: If the **Alchemy field name** you use within FormWare does not match an existing **Alchemy profile field name** in the IMR Alchemy database, a new profile field is created at the time of export. The new profile field name in the IMR Alchemy database is the **Alchemy field name** used in FormWare. The data type of the new profile field is text.

11. Select **OK** to accept your changes and return to the IMR Alchemy Configuration dialog. The FormWare field name and the Alchemy field name move from the **Non-Output Fields** list to the **Output Fields** list.

12. To edit a field, double-click on the field name in the **Output Fields** list and modify the **Alchemy field name** on the IMR Alchemy Field Mapping dialog. Select **OK** to return to the IMR Alchemy Configuration dialog.

13. To remove a field from the **Output Fields** list, highlight the field name and select the **Remove** button. The field name moves from the **Output Fields** list to the **Non-Output Fields** list.

14. To add or remove multiple fields from the **Output Fields** list on the IMR Alchemy Configuration dialog, press the [Shift] or [Ctrl] key and click with your mouse. Once you have selected multiple fields, select the **Add** or **Remove** button to add or remove the selected fields. When you have selected multiple fields to add, the FormWare field names are used by default as the **Alchemy field names**.

15. Select **OK** to accept your changes on this dialog and return to the Designer module.

Configuring a FileNET Export

The FileNET export component allows you to export data directly from FormWare into a FileNET repository. The following sections describe some differences between the FormWare and FileNET systems and how to set up FormWare to export data to a FileNET repository.

FileNET produces image management processing systems, which are referred to as document processing. FormWare has historically conducted forms processing. In document processing, the emphasis is on capturing just enough information off an image to identify the image and then to store the image with that information. In forms processing, the emphasis is on capturing as much information off the image as possible and then the image is discarded. With the support of FileNET export, FormWare can be used to conduct document-processing functions.

A *document* is a collection of one or more pages of information that are associated with one another as a group. In FormWare, a document is contained within a job and can contain a number of data forms, which can contain a number of fields. In FileNET, a document contains a number of fields, which may be spread across any or all pages that comprise the document.

Within FileNET, a *document class* is information about a document, such as number of pages, indexed fields, properties, etc. Typically, a FileNET document class corresponds to a single FormWare job. However, you can include multiple FormWare documents in a FormWare job, in which case each FormWare document within a job will correspond to a FileNET document class.

Note: A FormWare BDF can contain more than one document type, each being associated to a different FileNET document class. In FileNET, a batch can only contain documents associated with one document class. Because FileNET does not support more than one document class in a FileNET batch, a FormWare batch does not always translate into a single FileNET batch. When a batch containing different types of documents is exported from FormWare to FileNET, a FileNET batch is created for each of the different FormWare document types.

Understanding FileNET Export Requirements

Before you create a FormWare job for export to a FileNET system, you should finalize the design of both the FormWare job and the corresponding FileNET document class. This may take some planning time, to ensure that the fields in both the FormWare job and the FileNET document class conform to the same data type, length, and any other considerations. You will want to be cautious when assigning properties to the fields that could impede the export process. FormWare does not check during the design phase to verify that the FileNET document class properties are compatible with those in the FormWare job. Conflicts may only become apparent during export, such as exporting alphabetic data to a numeric only FileNET field.

When you configure a workstation to run the FileNET export or to run the **Tools > Configure> FileNET** menu option from the Administration module, you must have both IDM Desktop (version 3.2) and Panagon Capture Professional (version 3.1) correctly installed. Additionally, Panagon Capture must be configured to include the Commit component, Commit.ocx.

Verify that the export workstation can correctly log into the FileNET IMS prior to conducting a FileNET export from FormWare. This includes resolving network communications issues using TCP/IP or other means. You should run the Panagon Capture Logon.exe program to verify that the workstation can successfully log on to the correct IMS.

In addition, you must create an Export – FileNET Repository JobFlow task within JobFlow Designer to conduct an export session from FormWare to FileNET.

Setting System Properties for FileNET Export

Once you create a FormWare job, data form, fields, and JobFlow task for a FileNET export, you must configure the FormWare system for that export as well as the job and form. All levels of configuration are performed within the Designer module except system level configuration, which is done through the Administration module

The FileNET Export Configuration dialog contains the system-level configuration settings for the FileNET export component. These settings are required to establish a connection between a client workstation and an IMS server. To set these properties:

1. Open Administration.

2. Select the **Tools > Configure > FileNET** menu option. The FileNET Export Configuration dialog displays.

3. In the **Repository Type** field, select type of repository. Valid options are:

    o   **RILBes34.dll** – Used for export to IDMIS repositories.

    o   **RILLocal.dll** – Used for export to local repositories.

4. In the **Repository Name** field, select the name of the Panagon Capture session that is used whenever the FileNET Export component runs. This name is used to distinguish between multiple sessions that may be accessing a repository by defining a root prefix associated with items in the repository. This parameter is optional and is set to "FormWare" by default. For example, if this value is left to the default "FormWare", then a batch called "B000123" in a folder called "TestBatches" would be referenced as "FormWare:/TestBatches\B000123".

5. In the **Repository Connection (NCH name)** field, type the 3-part NCH resource name (OBJECT:DOMAIN:ORGANIZATION). This value is dependent on the value in the **Repository Type** field.

    o   If the **Repository Type** field is set to **RILBes34.dll**, then this value specifies the 3-part NCH name for the IDMIS repository. The name must include all three parts separated by colons. If the object part is not known (first part), then use "Bes" to indicate the default BES (FileNET Batch Entry Services).

    o   If the **Repository Type** field is set to **RILLocal.dll**, then this value specifies the data source name (DSN) for the local repository. This name must be configured as an ODBC data source and is set to "Local Repository" by the default Panagon Capture installation.

6. In the **Login Name** field, type the user name that the FileNET Export component uses to connect to the repository name.

7. In the **Login Password** field, type the user password that the FileNET Export component uses to connect to the repository.

8. Select the **Options** button to set the disposition and error disposition options. The FileNET Export Options dialog displays. If you set these options on this dialog, they will be system-level values that are relevant throughout the FormWare installation. Alternatively, you can configure these options at the job level, in which case they will override the system-level settings for that job.

9. In the **Disposition** and **Error Disposition** areas, select from the following values:

    o   **Index** - Exports batches to FileNET and sets the "Next Phase" to "Index."

    o   **Verify** - Exports batches to FileNET and sets the "Next Phase" to "Verify."

    o   **Commit** - Exports batches to FileNET and sets the "Next Phase" to "Commit." This means that the batch is not actually committed by FormWare—it is only marked as

"ready-to-be-committed." A subsequent processing step (performed by FileNET software) must be performed before the batches can be committed.

- o **Commit (async)** - Exports batches to FileNET and commits the batches asynchronously. The asynchronous method indicates that the FileNET export component adds the batch to the committal queue, but does not wait to see if the actual data was written successfully to disk. Batches committed asynchronously are not immediately available for retrieval by FileNET's query and retrieval software. Most applications (especially large volume ones) will choose asynchronous committal over synchronous due to its quicker processing time.

- o **Commit (sync)** - Exports batches to FileNET and commits the batches synchronously. The synchronous method means that the FileNET Export component adds batches to the committal queue and then waits to see if the actual data was written successfully to disk. This method is inherently slower than the committing asynchronously, however, it is potentially more reliable because the asynchronous method does not report errors that might arise between queuing a batch for committal and writing the data to disk.

- o **Delete** - Deletes batches with errors from the BES. Do not use this setting except while testing an application or for batches within a production environment that have errors. This option is only available for **Error Disposition**.

10. Select the **Busy Override** check box to disable the busy override processing of a batch. When the **Busy Override** option is turned off, batches with a "Busy" status are not processed by the FileNET export component. The default setting is to automatically override busy batches and process them.

11. Select **OK** to return to the FileNET Export Configuration dialog.

12. Select the **Test** button to perform a test connection to the configured FileNET repository. The current properties are used to attempt the connection, and the success (or failure) of the connection is indicated in a subsequent dialog.

13. Select **OK** to accept your settings and return to Administration.

Setting Job Properties for FileNET Export

The FormWare job you create must contain fields corresponding to every indexed field within the FileNET document class to which you want to export data. You can create fields other than the indexed fields to capture additional data for use in the FormWare system. However, only data captured by the indexed fields are exported to the FileNET repository.

To set FileNET Export job properties:

1. Open a job in Designer.

2. Display the Job Properties, General tab for the job.

3. In the **Component** field, select the **FileNET Export** option.

4. Select the **Configure** button to display the FileNET Export Job Properties dialog.

5. In the **Document Class Name** field, select the **Browse** (...)button to find an existing document class name to use for this job. Each job should be associated with a different FileNET Document Class. If you leave the **Document Class Name** field blank, the system uses the **Document Name** specified on the Form Properties, Edits tab in Designer.

6. Select the **Options** button to display the FileNET Export Options dialog.

7. In the **Disposition** and **Error Disposition** areas, select from the following values:

    o   **Index** - Exports batches to FileNET and sets the "Next Phase" to "Index."

    o   **Verify** - Exports batches to FileNET and sets the "Next Phase" to "Verify."

    o   **Commit** - Exports batches to FileNET and sets the "Next Phase" to "Commit." This means that the batch is not actually committed by FormWare—it is only marked as "ready-to-be-committed." A subsequent processing step (performed by FileNET software) must be performed before the batches can be committed.

    o   **Commit (async)** - Exports batches to FileNET and commits the batches asynchronously. The asynchronous method indicates that the FileNET export component adds the batch to the committal queue, but does not wait to see if the actual data was written successfully to disk. Batches committed asynchronously are not immediately available for retrieval by FileNET's query and retrieval software. Most applications (especially large volume ones) will choose asynchronous committal over synchronous due to its quicker processing time.

    o   **Commit (sync)** - Exports batches to FileNET and commits the batches synchronously. The synchronous method means that the FileNET Export component adds batches to the committal queue and then waits to see if the actual data was written successfully to disk. This method is inherently slower than the committing asynchronously, however, it is potentially more reliable because the asynchronous method does not report errors that might arise between queuing a batch for committal and writing the data to disk.

    o   **Delete** - Deletes batches with errors from the BES. Do not use this setting except while testing an application or for batches within a production environment that have errors. This option is only available for Error Disposition.

8. Select the **Busy Override** check box to disable the busy override processing of a batch. When the Busy Override option is turned off, batches with a "Busy" status are not processed by the FileNET export component. The default setting is to automatically override busy batches and process them.

9. Select **OK** to accept your changes and return to the FileNET Export Job Properties dialog.

10. Select **OK** to accept your changes and return to the Job Properties dialog.

Setting Form and Field Properties for FileNET Export

Once you have completed any job configurations needed for FileNET export, you can configure the form and field options. Field configuration is also performed at the form level, providing a complete picture of all the fields being exported to FileNET.

To set form and field level properties for FileNET export:

1. Open a job and form in Designer.

2. Display the Form Properties, General tab for the form.

3. In the **Component** field, select the **FileNET Export** option.

4. Select the **Configure** button to display the FileNET Export Form Properties dialog.

5. In the **Exported Image** area, select the type of images you want to export from the FormWare to FileNET. Valid options are:

    o   **Original** - Exports original images (as referenced in the BDF) to the FileNET system. This is the default setting.

o **Enhanced** - Exports enhanced images that result from running the enhancement process to the FileNET system.

o **None** - Exports no images to the FileNET system. However, document data (index field values) are still exported.

o **Color** - Export color images (JPG files) to the FileNET system. Select this option if you are processing JPG images.

6. Select the **Add** button to add fields for exporting. The Select Field dialog displays a list of all the fields on the current form.

7. Double-click the field you want to add to the list of fields to export. The FileNET Export Field Properties dialog displays for the selected field.

8. In the **FileNET Index Name** field, type the name for this FormWare field within the FileNET system. This field enables you to have different names for your FormWare fields and their corresponding FileNET fields.

9. Select the **Trim Left** or **Trim Right** check boxes to clean up the exported data before it is sent to the FileNET system. Trim Left removes any white space from the left side of the field, while Trim Right removes white space from the right side of the field. In both cases, this includes the removal of spaces and tab characters.

10. Select **OK** to accept the changes and return FileNET Export Form Properties dialog.

11. Continue adding fields until you have indicated all the fields necessary for FileNET export.

12. To modify a field, highlight it and select the **Edit** button. Make your changes and select **OK**.

13. To remove a field, highlight it in the list and select the **Remove** button.

14. Select **OK** to accept your changes and return to Designer.

15. Select **Save All** on the Designer window toolbar to save your settings.

Accessing the FileNET Export Document ID Using VBA

You can use a call with a VBA project to retrieve the document ID number (DocID) assigned by FileNET Export. To retrieve the FileNET-assigned DocID, you must run a VBP within the context of the FileNET Export client (i.e., insert -VBP=xxx.vbp on the Client Parameters line for the FileNET Export task in either WorkFlow or JobFlow Designer). Within the VBP, you can access the DocID within an OnDocEnd event by calling docID = Int(GetProperty("FNDOCID")). This property is only valid in the OnDocEnd event (i.e., after the document has been exported to FileNET and the DocID value is known).

Following is a syntax example:

```
Dim docID as integer
docID = Int(GetProperty("FNDOCID"))
```

Note: The DocID is assigned by FileNET's APIs during the export process, so the DocID is not available during OnFormStart/End or OnImageStart/End. Data has already been exported prior to OnDocEnd; therefore there is no way to export the DocID with the data being exported. All the images and data for a document get physically exported between the last OnImageEnd for the document and the OnDocEnd.

Configuring a Unisys InfoImage (RBE) Export

The Unisys InfoImage Rapid Batch Entry (RBE) export component allows you to export index data and images directly from FormWare into a Unisys InfoImage workflow system. Documents submitted to an InfoImage RBE repository using this process can be employed for archival, entry into the InfoImage workflow, or both. Once successfully submitted into InfoImage, the images are available for query, retrieval, printing and routing by end users of the InfoImage system.

Before you create a FormWare job for export to an InfoImage RBE export system, you should finalize the design of both the FormWare job and the corresponding InfoImage RBE document class. This may take some planning time to ensure that the fields in both the FormWare job and the InfoImage RBE document class conform to the same data type, length, and other considerations. Use caution when assigning properties to the fields that could impede the export process. FormWare does not check during the design phase to verify that the InfoImage RBE document class properties are compatible with those in the FormWare job. Conflicts may only become apparent at the time of export, such as exporting alphabetic data to a numeric-only InfoImage RBE field.

Understanding InfoImage RBE Output Files

The InfoImage RBE export process generates Data Definition files (DDF) and acknowledgement files to send to InfoImage. The DDF (or control) file serves as a table of contents for the different files in a directory for a given batch. It specifies the grouping of images and other files as well as the name and class of each document. The InfoImage RBE export process creates one DDF for each batch data file, containing up to 1,000 documents. A single FormWare batch data file generates a DDF with the same name as the batch data file, but with a .ddf extension.

In addition to the DDF, a status or acknowledgement file is also generated during export. The status file is used to communicate between FormWare and InfoImage. There are three zero length status files used to determine the transmission state of a DDF: Submit.sts, Finished.sts and Error.sts. These names are static, and do not change with each batch. The FormWare RBE export process creates a Submit.sts file once a DDF is successfully generated. This is an indication to the InfoImage system that a DDF file is ready to upload. If the DDF file is uploaded successfully, InfoImage replaces Submit.sts with Finished.sts. If the DDF file upload is unsuccessful, InfoImage replaces Submit.sts with Error.sts. If an Error.sts file is generated, the unsuccessful batch can be manually routed back to the RBE export task to attempt to re-send the DDF file to the InfoImage system. The manual routing of batches within FormWare is performed in JobFlow maintenance.

The version 4 DDF format is used for the InfoImage RBE export process. A DDF file includes batch, document, and page records. The batch records precede all document records in the DDF; and one or more groups of document records follow the batch records. Each group of document records describes the content of every document in the batch. The DOCUMENT delimiter separates each document. If page records exist, they follow each group of document records.

See the following sample DDF. The sample batch (BATCH1) contains two documents. Each document contains two image pages and two index fields. All documents are class HCFA1500. The resulting DDF, Batch1.ddf, would look as follows:

```
VERSION|4
CLASS|HCFA1500

DOCUMENT
DOCNAME|Batch1-Doc1
DOCFIELD|CLAIMNUMBER|1-1
```

```
DOCFIELD|RECEIPTDATE|1/1/99
PAGE|G:\FORMWARE\IMAGES\PAGE111.TIF
PAGE|G:\FORMWARE\IMAGES\PAGE112.TIF
DOCUMENT
DOCNAME|Batch1-Doc2
DOCFIELD|CLAIMNUMBER|1-2
DOCFIELD|RECEIPTDATE|1/2/1999
PAGE|G:\FORMWARE\IMAGES\PAGE121.TIF
PAGE|G:\FORMWARE\IMAGES\PAGE122.TIF
```

The parameters used in DDF files are described below.

Batch Record Parameters

Batch record parameters govern all document record parameters in the DDF.

Record Syntax	Description	Note
VERSION|4	Version number for DDF format.	REQUIRED
CLASS|doc-class	Default document class. All documents in the batch are assigned this class, except those specified in a set of DOCUMENT records.	REQUIRED: The default name is defined by the Document Name on the Form Properties, Edits tab .
AUTOICON|y|n	Specifies whether a page icon is generated for each page in the file that does not have an associated PAGEICON record. An "n" means no page icon, a "y" means generate a page icon.	NOT REQUIRED. Only included in the DDF if defined as a property on the Unisys InfoImage Properties dialog.
NUM_DOC|number	Number of documents in the batch. Batch verification will fail if not equal to the actual number of documents in the file. An optional crosschecking method for integrity of batch data in the file.	NOT REQUIRED. Only included in the DDF if defined as a property on the Unisys InfoImage Properties dialog.
MAX_FIELDS|number	Maximum number of field records per document. Batch verification will fail if not greater than or equal to the actual number of fields for any associated documents in the file. An optional crosschecking method for integrity of batch data in the file. May occur multiple times per file. Applies to all documents whose opening delimiter occurs after the first MAX_FIELDS record and before the next MAX_FIELDS record, if any, in the file.	NOT REQUIRED. Only included in the DDF if defined as a property on the Unisys InfoImage Properties dialog.

Document Record Parameters

Following the batch records are groups of document records that describe the content of each document in the batch. The DOCUMENT delimiter separates each document.

Record Syntax	Description	Note
DOCUMENT	Document delimiter.	REQUIRED: Precedes each document in the file.
DOCNAME|name	Document name.	REQUIRED: Default value is the first field on the form. Use the Unisys InfoImage Properties dialog to specify an alternate field value.

Record Syntax	Description	Note
DOCFIELD|name|value	Form field name and value. Multiple DOCFIELDs are allowed per document, each with a different name.	REQUIRED: The names and values of the FormWare fields marked for RBE Export.
NUM_PAGE|number	Number of pages in the document. Batch verification will fail if not equal to the actual number of pages in the file for this document. An optional cross-checking method for integrity of batch data in the file.	NOT REQUIRED. Only included in the DDF if defined as a property on the Unisys InfoImage Properties dialog.
CLASS|doc-class	Document class for this document. Overrides the CLASS batch record as described above.	NOT REQUIRED. Only included in the DDF if defined as a property on the Unisys InfoImage Properties dialog.

Page Record Parameters

These records describe the parameters for the individual pages in the document.

Record Syntax	Description	Note
PAGE|path|name	Path and name of image page TIFF file.	The default is to indicate the full path and name of the TIFF file. The "name" parameter is only included in the DDF if defined as a property on the Unisys InfoImage Properties dialog.

By default, the InfoImage RBE export process stores the DDF and acknowledgement files in the \FormWare\Results\RBEClnt\<batchname> directory. During export, a directory is created for each batch, using the name of the batch as the name of the directory. You can change the default location by editing the Output Directory field in the Unisys InfoImage Properties dialog. The Output Directory indicates the path name to the base directory where each export batch's sub-directory is created.

If you choose to change the default location of the InfoImage RBE export files, please note the base directory location must exist or be created prior to attempting to export batches from FormWare. The only directories created by the RBE export process are the batch name sub-directories that reside under the specified base directory.

Configuring InfoImage RBE Export

The FormWare job you create must contain fields corresponding to every index field within the InfoImage RBE document class to which you want to export data. You can create fields other than index fields on your FormWare forms. These fields can be used to capture data for export to other locations such as a database or text file. Only the fields configured for RBE export are referenced in the DDF file.

Once you have created a FormWare job, data form, fields, and JobFlow task for an InfoImage RBE export, you must configure the FormWare system for that export. To allow FormWare to export to an InfoImage RBE repository, you must configure it at the form level. To configure a form for InfoImage RBE export:

1. Open a job and form in Designer.

2. Display the Form Properties, General tab for the form.

3. In the Component field, select the Unisys InfoImage (RBE) Export option.

4. Select the **Configure** button to display the Unisys InfoImage Configuration dialog. By default, all fields on the form display in the **Non-Index Fields** area and are referenced by their FormWare field name.

5. Highlight a field in the **Non-Index Fields** list and select the **Add** button to display the Unisys InfoImage Field Mapping dialog.

6. In the **Enter mapping field name** field, type the field name as you want it to appear in the DDF. The name you type should be similar to the corresponding field name in the InfoImage RBE Repository.

Note: Do not use spaces in field names. Use the underscore character (_) instead of a space.

7. Select **OK** to accept your changes and return to the Unisys InfoImage Configuration dialog. The new index field displays in the **Index Fields** list with both the original FormWare field name and the new InfoImage index field name in parentheses.

8. To remove a field from the **Index Fields** list, highlight the field and select the **Remove** button. The field is removed from the **Index Fields** list and added to the **Non-Index Fields** column referenced by its FormWare field name.

9. If you have assigned a **Document Name** value on the Form Properties, Edits tab, the **Properties** button becomes active on the Unisys InfoImage Configuration dialog. Select the **Properties** button to display the Unisys InfoImage Properties dialog. See to the Batch Record Parameters section in Understanding InfoImage RBE Output Files for more information about the fields on this dialog.

10. In the **Default document class** field, type the document class name to be used for all documents in the batch. The default name is defined by the **Document Name** setting on the Form Properties dialog, Edits tab.

11. In the **Document name (DOCNAME) field**, select the field name to use for the document name. The data from this field is used to designate the DOCNAME in the DDF. The default field is the first field in the form.

12. In the **Output Directory** field, select the **Browse** button to locate the path to the directory where files should be exported.

13. In the **Optional Records** area, define the following options:

    a. Select the **Document specific class** check box to activate the **Document class field**, where you can select a FormWare field whose data should be used to designate a document class that overrides the default document class. This option allows you to specify multiple document classes per batch.

    b. In the **Document class field**, select a FormWare field to assign as the document class field.

    c. Select the **Auto Generate Page Icon** check box to specify the **AUTOICON** value that is written to the DDF. By default, the **AUTOICON** line does not appear in the DDF. If you choose to have the line included, then select **Yes** or **No** in the **AUTOICON** area.

    d. Select the **Number of documents in batch** check box to include the specified number of documents in the DDF. This value can be used to determine batch integrity within the InfoImage system.

    e. Select the **Maximum number of field records per document** check box to write the maximum number of field records per document to the DDF. This value can be used as a cross-check validation for the batch once it is in the InfoImage system.

f. Select the **Number of pages in this document** check box to indicate the number of pages per document that should be written to the DDF. This value can be used for validation within the InfoImage system.

g. Select the **Name of image page TIFF file** check box to indicate that the additional "name" parameter should be included in the page records in the DDF. This activates the **Default page name** and **Page name field** fields

h. In the **Default page name** field, type the page name for any page (image) that does not include a field of the name selected as the page name field.

i. In the **Page name** field, select the field whose data should be used to determine the page name. The default page name is used.

14. Select **OK** to accept your settings and return to the Unisys InfoImage Configuration dialog.

15. Select **OK** to return to the Form Properties dialog.

16. Select **Save All** on the Designer window toolbar to save your changes.

Configuring an XML Export

XML (extensible markup language) is a markup language for documents containing structured information, including text and graphic content, and the constraints or rules that indicate what role the content plays. Similar to HTML, XML uses tags (text bracketed by < >) and elements (data between tags). Unlike HTML, XML does not specify what each tag means or how the text between tags will appear in a web browser. XML uses tags only to delimit pieces of data and leaves the interpretation of that data to the application reading it.

The XML export process allows you to export data to an XML-formatted, flat text file. The export component will generate one output file per transaction or batch. The filename is the same as the FormWare BDF file name, but with a .xml extension and resides in the same directory as the BDF.

Understanding XML Output Files

The XML export process exports all data from a BDF and the specified level of data from the associated images to an output file. Data is output as one file per transaction. The following diagram illustrates the transaction structure of an XML output file: UNREADABLE
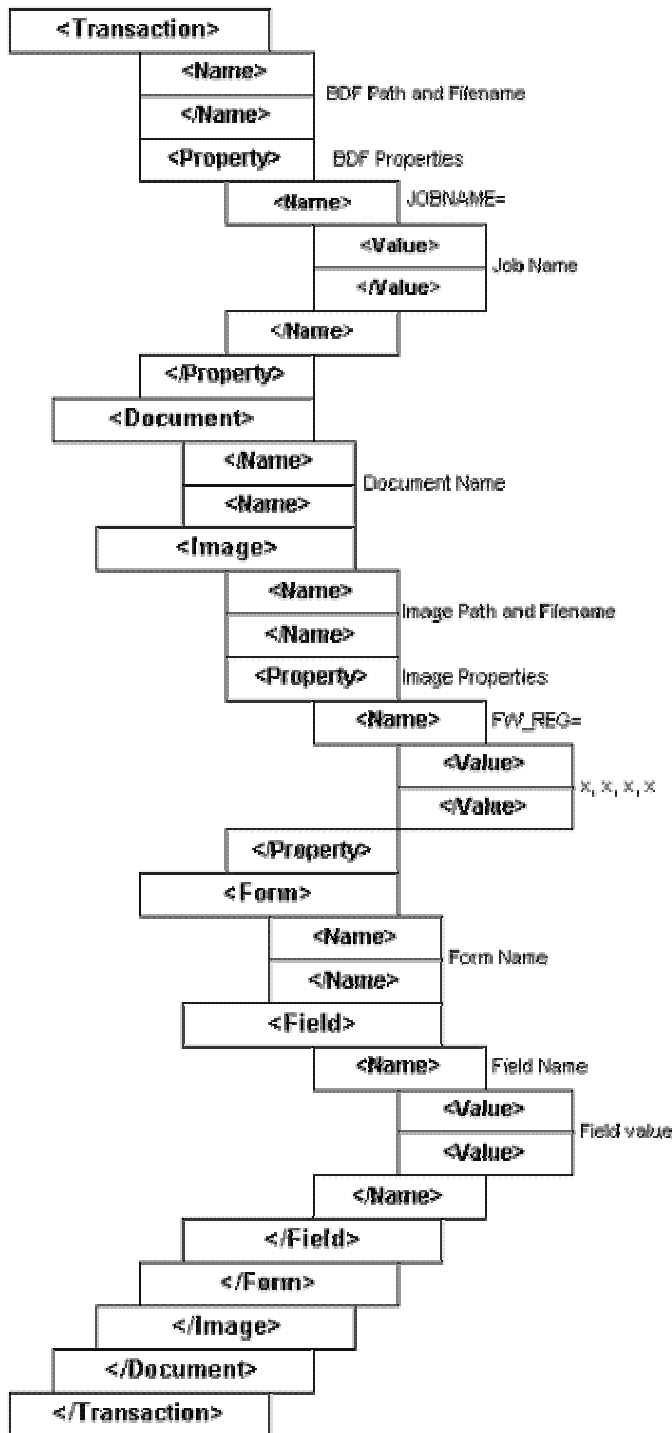
Following is a sample BDF and the resulting XML file after running the XML export process. The data in the XML file is a combination of information from the BDF and data captured from the images referenced in the BDF. The captured data resides in the batch data file created during Completion.

```
JOBNAME=CAPTURE
C:\FORMWARE\IMAGES\DEMO0001.TIF
FW_REG=0,0,2.2,5
C:\FORMWARE\IMAGES\DEMO0002.TIF
C:\FORMWARE\IMAGES\DEMO0003.TIF
```

The following is an example of the 1040ez.xml file. This XML file resides in the same location as the BDF (e.g., \FormWare\Files).

```xml
<?xml version="1.0" ?>
- <transaction>
    <name>c:\formware\files\1040ez.bdf</name>
  - <property>
      <name>JOBNAME</name>
      <value>capture</value>
    </property>
  - <image>
      <name>C:\FORMWARE\IMAGES\demo0001.tif</name>
    - <form>
        <name>CAPTURE</name>
        <CAPTURE_Fld001>JOHN JACKSON</CAPTURE_Fld001>
        <CAPTURE_Fld0002>MARY JACKSON</CAPTURE_Fld0002>
        <CAPTURE_Fld0003>NEW ORLEANS</CAPTURE_Fld0003>
        <CAPTURE_Fld0004>LA</CAPTURE_Fld0004>
        <CAPTURE_Fld0005>70115</CAPTURE_Fld0005>
        <CAPTURE_Fld0006 />
        <CAPTURE_Date />
      </form>
    </image>
  - <image>
      <name>C:\FORMWARE\IMAGES\demo0002.tif</name>
    - <property>
        <name>FWREG</name>
        <value>0,0,2.2,5</value>
      </property>
    - <form>
        <name>CAPTU001</name>
        <CAPTU001_Fld0001>000123455547</CAPTU001_Fld0001>
      </form>
    </image>
  - <image>
      <name>C:\FORMWARE\IMAGES\demo0003.tif</name>
    - <form>
        <name>CAPTU001</name>
        <CAPTU001_Fld0001>000000032544</CAPTU001_Fld0001>
      </form>
    </image>
  </transaction>
```

Setting Job Properties for XML Export

At the job level, you define tag mapping and indicate whether data for each tag level is to be output during the export process. To set properties for an XML export at the job level:

1. Open a job in Designer.

2. Display the Job Properties, General tab for the job.

3. In the **Component** field, select the **XML Export** option.

4. Select the **Configure** button to display the XML Properties dialog.

5. In the **Output Directory** field, select the **Browse** button to select the path to the directory where the XML files are to be exported. The default location is same as the location where BDFs are processed by the FormWare system.

6. Select the **Use field names as tags** check box if you want to use the field names on the data form as tags. By default, the values in the **FormWare XML Tag** and **User XML Tag** columns are the same.

Note: If you select the **Use field names as tags** option, verify that all fields on the form begin with an alpha character. If some fields on the data form begin with numeric or special characters, the XML export process cannot use them as tags and will display an error message.

7. To change the **User XML Tag** value for any **FormWare XML Tag**, highlight the tag and select the **Edit** button. The XML Tag Mapping dialog displays.

8. In the **User XML tag** field, type a new name. The **User XML tag** name must begin with an alpha character. The new name should conform to your own terminology that describes a similar layer or hierarchy level. You cannot change the **FormWare XML tag**.

9. Select the **Output this tag level** check box to indicate that the data with the tags at this level are included in the output. This check box is selected by default; to exclude the tag and its associated data at this level, remove the check mark. The transaction, property, name, and value level tags are required for output. Therefore, the Output this tag level option cannot be unchecked at those levels.

10. Select **OK** to accept your changes return to the XML Properties dialog.

11. Select **OK** to return to the Job Properties dialog.

12. Select the **Save All** button on the Designer window toolbar to save your settings.

Setting Form Properties for XML Export

At the form level, you will indicate whether each field and its data on your form will be output during the export process. To set properties for XML export at the form level:

1. Open a job and form in Designer.

2. Display the Form Properties, General tab for the form.

3. In the **Component** field, select the **XML Export** option.

4. Select the **Configure** button to display the XML Configuration dialog. By default, all fields on the form are included in the **Output Fields** list.

5. To remove a field from the **Output Fields** list, highlight the field and select the **Remove** button.

6. To add a previously removed field back to the **Output Fields** list, highlight the field and select the **Add** button. The XML Field Mapping dialog displays.

7. In the **Enter mapping field name** field, type the new name for this field within the XML export file. This does not change the name of this field within FormWare.

Note: Do not use spaces in this field name; spaces are not acceptable in XML fields. Instead, use the underscore character (_).

8. Select **OK** to accept your changes and return to the XML Configuration dialog.

9. Select **OK** on the XML Configuration dialog to validate your changes and return to the Form Properties dialog.

10. Select **Save All** on the Designer toolbar to save your settings.

Configuring an Optika eMedia Export

The Optika eMedia™ export component allows you to export index data and images from FormWare into the Optika eMedia document management system. Documents submitted to Optika® using this process can be employed for archival and retrieval at a later point. Electronic report management, transaction viewing, and workflow processing are all facilitated through the Optika eMedia document management system.

Understanding Optika eMedia Output Files

The Optika eMedia export process translates a user-defined set of batch data file information into the format required by eMedia. One file is output for each document type in a BDF. The name of the resulting export file is the same as the batch from which it is derived with a configurable extension. The file extension defaults to the first three characters of the current form name. The file extension is intended to distinguish between document classes; the form name is used to derive the extension. By default, the Optika eMedia export process stores the export files in the \FormWare\Results\OptClnt directory.

Optika eMedia does not support long file names, and therefore requires that all filenames be eight or fewer characters, with a three-character extension. Because FormWare can use batch data file names up to 27 characters long, the Optika eMedia export will truncate the export file name if a long file name is used. To ensure unique file names during the export step, adhere to this restriction.

The eMedia Filer default format is used for the Optika eMedia export process. The structure of the format is:

```
<ObjectName>|Field 1|Field 2|Field 3|…|Field n
```

There is one line in the export file for each object containing associated data. ObjectName refers to an image file name and must include the complete drive designation and path to the image. Be aware that UNC path names (\\ServerName\…) are not supported by Optika eMedia.

For example, the sample batch (DEMO) contains two forms, each representing a different document class. The first form is configured to export three index field values and create a file with a .rev extension. The second form is configured to export two index field values and create a file with a .inc extension. The batch, Demo.bdf, creates two output files named Demo.rev and Demo.inc.

Demo.rev contains a single line:

```
C:\FORMWARE\DBINFO\DEMO0001.TIF|JOHN JACKSON          |LA|123778444
```

Demo.inc contains a single line:

```
C:\FORMWARE\DBINFO\DEMO0002.TIF|DAN BURNETTE          |224905477
```

Before you create a FormWare job for export to an Optika eMedia document management system, finalize the design of both the FormWare job and the corresponding eMedia document class. This may take some planning time, to ensure that the fields in both the FormWare job and

the Optika eMedia document class conform to the same data type, length, and other considerations. Be cautious when assigning properties to the fields that could impede the export process. FormWare does not check during the design phase to verify that the Optika eMedia document class properties are compatible with those in the FormWare job. Conflicts may only become apparent at the time of export.

Configuring the Optika eMedia Export

The FormWare job you create must contain fields corresponding to every index field within the Optika eMedia document class to which you want to export data. You can create fields other than index fields on your FormWare forms. These fields can be used to capture data for export to other locations such as a database or text file. Only the fields configured for eMedia export are referenced in the export file created by the Optika eMedia export process.

Once you have created a FormWare job, data form, fields, and JobFlow task for an Optika eMedia export, you must configure the FormWare system for that export. You must configure FormWare at the form level to export to an Optika eMedia document management system. Follow these steps to configure FormWare for Optika eMedia export:

1. Open a job and form in Designer.

2. Display the Form Properties, General tab for the form.

3. In the **Component** field, select the **Optika eMedia Export** option.

4. Select the **Configure** button to display the Optika eMedia Configuration dialog. By default, all fields on the form display in the **Non-Index Fields** list and are referenced by their FormWare field name unless you selected the **Indexed Field** option on the Field Properties, Checkbox Edits tab for any of the fields. If you have selected the **Indexed Field** option for a particular field, that field displays in the **Index Fields** list when you open the Optika eMedia Configuration dialog.

5. To add a field to the **Index Fields** list, highlight a field in the **Non-Index Fields** list and select the **Add** button. The field is moved to the **Index Fields** list and retains its FormWare field name.

Note: The Optika eMedia file format does not contain tag values or field names for the index field data. Therefore, you cannot alter the field names within Optika eMedia as you can with other FormWare export components.

6. To remove a field from the **Index Fields** list, highlight the field and select the **Remove** button. This moves the field back to the **Non-Index Fields** column.

7. To change the default input directory location and file extension used in the Optika eMedia export process, select the **Properties** button to display the Optika eMedia Properties dialog.

8. In the **eMedia Filer input directory field**, select the **Browse** button to locate the full path to the directory where export files will be created for the current form type. Although the drive letter must refer to a local drive or be mapped to a logical drive, the directories are created by the Optika eMedia export component if they do not already exist.

9. In the **eMedia Filer input file extension** field, type the three-character extension to use when exporting files for the current form type. The default file extension is the first three characters of the current form name.

> **Note**: The file extension used to distinguish between different document classes. If your form names do not reflect a document class within the first three characters, you should consider changing the extension value.

10. Select **OK** to accept your changes and return to the Optika eMedia Configuration dialog.

11. Select **OK** on the Optika eMedia Configuration dialog to validate your changes again and return to the Form Properties dialog.

12. Select **Save All** on the Designer window toolbar to save your changes.

Configuring an FPL Export

FPL export refers to exporting batch data using a program created with the FormWare Programming Language (FPL). The following sections provide you with information about setting up and performing an export process using FPL.

When exporting data using FPL, you must use an export program and an export device. The export program contains source code that formats the exported data. The device points to the text file or printer where the data is exported. FPL export programs are executed against one or more batch data files. The program file has an .et extension and resides in the \FormWare\Export folder.

> **Note**: The Export folder is referenced as the Output folder in MS Windows Explorer; therefore a correct pathname includes "Output" not "Export".

Writing an FPL Export Program in Administration

Follow these steps to create an export program within the Administration module:

1. On the Administration window, display the contents of the Export folder.

2. Right-click in the Contents area and select **New** from the context menu. The text editor displays.

3. Type the source code, written in FPL, for your export program. Refer to the FormWare Programming Reference Guide for more information on FPL.

4. Select **Save** within the text editor to save your new export program.

5. Type the name of the new export program and save the program file.

6. **Exit** the text editor. The new export program displays in the Export folder Contents area with an .et extension.

7. To modify an export program, double-click on the ET file in the Export folder Contents area. This launches the text editor and displays the source code. Edit and save your changes to the program file.

Once you have created and saved the new FPL export program, you are ready to parse the program. Parsing a program converts the syntax of its source code to a binary file, which is then used at run-time. The binary file has the same name as the corresponding program file but has a .r extension.

Follow these steps to parse an export program within the Administration module:

1. On the Administration window and display the contents of the Export folder.

2. Highlight an ET program in the Contents area, right-click to display the context menu and select **Compile**. This parses the export program and creates the .r file.

3. To re-parse an existing program, select the **Show All Files** button and then double-click on the .r file. You can also parse any export program associated with a job within the Designer module.

Writing an FPL Export Program in Designer

You can specify an export program in a job, using the Designer module. When you conduct an FPL export of the batch data file created by the job, the export program specified in that job is used by default. The job export properties information is maintained in the batch data file header.

You can also conduct an FPL export of batches without using the export program specified in the job. The procedure for this depends on the export method you are using. Refer to the appropriate section of Export Methods for information on using job export properties or overriding them during an export session.

Follow these steps to create an export program within the Designer module:

1. Open the job in Designer.

2. Display the Job Properties, Export tab.

3. Select the **Browse** button next to the **Export Program** field. The Browse Files dialog displays.

4. To create a new export program, type the name in the **File Name** field instead of selecting an existing program. You do not need to include the .et file extension; it is added automatically when the file is created.

5. Select the **Open** button and an export program with the name you specified is created and displays in the **Export Program** field.

6. Select the **Edit** button next to the **Export Program** field to display the program in the text editor.

7. Type the program syntax for the export program in the editor.

8. Save and exit the editor.

9. In Designer, display the Job Properties, Export tab. An export program should be specified in the **Export Program** field.

10. Select the **Compile** button next to the **Export Program** field. A progress dialog displays while the program is being parsed. This parses the export program and creates the .r file.

Using FPL as an Export Language

You can write export programs in FPL. See the FormWare Programming Reference Guide for instructions on using FPL.

The OUTPUT and TYPE verbs are the most commonly used commands to export data from the FormWare system. However, the System OS commands offer some additional flexibility in how the data is exported.

Output vs. Type Verb

When you write source code for an export program, you use either the *output* or the *type* verb to create the export program. A device associated with an export program that contains *output* statements is referred to as an *output* device, while a device associated with an export program that contains *type* statements is referred to as a *type* device. A device can behave differently depending on whether it is an output or type device.

The device called depends on which type of statement you use. An output statement sends data to the output device specified in the job. A type statement sends data to the type device specified in the statement. The syntax for a type statement is:

```
type <loc ##> data
```

where ## represents the type device number with a range of 01-99. When you create a device for type statements, you must name it "type##" where ## represents the device number. The type statement will send data to the type device with the corresponding number. For example, if the type statement is "type <loc 04> data", then data is sent to a device named TYPE04.DEV.

You place the defer control function at the end of an output statement. When this command is encountered, the next output statement is continued on the same line.

Change Out System Verb

The *change out* system verb changes the export device during export program execution. The [APPEND] option appends data to an existing data file. The following is an example:

```
system "@CHGOUT [APPEND] path\file name" gives retval
```

Change Type System Verb

The *change type* system verb changes the type device during export program execution. The [APPEND] option appends data to an existing data file. The following is an example:

```
system "@CHGTYP [APPEND] path\file name" gives retval.
```

Example Export Programs

The following are examples of export programs and the data they produce. These are simple programs using the OUTPUT verb and based on the following data:

Field Number	Field Length	Field Data
1	7	1367598
2	10	JENKINS
3	10	DENISE
4	20	421 MARKET STREET
5	10	MINNETONKA

Exporting Using the <all> Control Function

The following is an example export program that uses the <all> control function.

```
output <all>.
release.
```

Sample data:

```
1367598JENKINS   DENISE    421 MARKET STREET    MINNETONKA
```

Exporting Field Data

The following is an example export program that exports data by field. The number (#) represents the entire field contents of the field number indicated inside the parentheses. You can use a variable in place of the actual field number.

```
output (3)(5).
release.
```

Sample data:

```
DENISE    MINNETONKA
```

Exporting Field Sub-Positions

The following is an example export program that exports sub-positions within a field. The (#:m-n) represents field position m through n for the indicated field number.

```
output (3:1)(2)(1:4-6).
release.
```

Sample data:

```
DJENKINS   759
```

Exporting Field Literals

The following is an example export program that exports literal values along with field data.

```
output 'Name: '(2)','(3).
release.
```

Sample data:

```
Name: JENKINS   ,DENISE
```

Defining Export Devices

Devices point to the location of the exported data. An export device contains the path and filename of the text file where the data is exported. You can create a new text file for each export session or you can append the data to an existing file. The device also contains properties that impact the behavior of the export process.

During an FPL export process, data is exported to a device. A device allows a file to be created repeatedly without having to specify the full path and filename. You can associate an export device with a job. In this way, the properties set for the device can be applied to the batch data files created using that job.

Creating Export Devices

To create an export device within the Designer module:

1. Open a job in Designer.

2. Display the Job Properties, Export tab for the job.

3. Select the **Browse** button next to the **Export Device** field to display the Browse Files dialog.

4. To create new export device, type the name of the new device in the **File name** field. You do not need to include the .dev extension; it is added to the filename when the device file is created. The device name is restricted to eight characters. Export devices can behave differently depending on whether you use the OUTPUT or the TYPE verb in the

corresponding export program. Devices used with the TYPE command must be named "TYPE##.DEV" where ## represents a number in the range of 01-99.

5. Select the **Open** button and the export device with the name you specify is created and displays in the **Export Device** field on the Job Properties, Export tab.

6. Select the **Edit** button next to the **Export Device** field to display the export device in the Properties dialog, where you assign properties to the device.

7. In the **Path Name** field, type the full path and file name (with extension) of the text file to create when this device is called during an export session. This destination can be either a text file or a printer. If you specify a text file in the **Path Name** field, then a text file with the specified name is created and the exported data is written to it. The text file is not created until the OUTPUT or TYPE command is executed during an export session.

8. If you specify the pathname and filename of a printer in the **Path Name** field, the exported data is sent to that printer. To specify a printer, type LPT1 or the network printer name. You cannot specify a device name that has the name LPT1 and an extension as a file. LPT1, LPT2, LPT3, and LPT4 are reserved and are assumed to be print devices. The names in the path name cannot be more than eight characters; although the entire path name can be more than eight characters.

9. In the **List Name** field, type the name of the text file in the **Path Name** field with a batch list file (.blf) extension. Use this field if you are using the List Options function in your export process. The JobFlow module uses the **List Name** field and **List Options** settings to group batches together for export. The file in the **List Name** field acts as a monitor of this grouping. The values in the **List Options** determine when that group of batches is exported.

10. The file in the **List Name** field is created with a .blf extension in the location specified. Batches sent to the export device are then listed in this BLF file. When the specified Count or Time is reached, all of the batches listed in the BLF file are exported together and the BLF file is deleted. This allows JobFlow to group batches together for export.

Note: You can replace part of the path in the **Path Name** and **List Name** fields with the parameter %FORMWARE%. If you specify this parameter, the value specified by the FORMWARE environment variable is used for the path.

---

11. In the **Process** field, type the name of the program to be executed at the end of the export session. Some programs can be used to view the exported data or can be a communications package or other gateway software to upload the data to a mainframe computer. Process applies to OUTPUT devices only.

Note: When testing your forms processing application, we recommend that you type FWEDIT in the **Process** field. The exported data is then displayed in a text editor when the export session is completed. This is not advised for production environments. The VIEWDATA.DEV file is an example of a device used to view exported data.

---

12. Select from the following check boxes in the **Device Options** area, to control how data is written to the export device:

    o **Append to Device** - Append data exported to this device to the file specified in the **Path Name** field, if the file exists. If the file does not exist, it is created and data is written to it.

    o **Mark as Output** – Modify the batch data file header information after an FPL export session to mark the batch as having been exported. The **Export** option of the job export protections searches the batch header for this information to determine whether the batch has been exported.

- o **Out Extension** – Write output statements in the export program to a file with the batch name and an .out extension. If this option is selected, the **Path Name** field should not include a filename. This is for Output devices only.

- o **Print Device** - Append a <CR><LF> (a carriage return and line feed) to the end of each output statement when exporting data. This means that the data will be broken into many lines.

- o **Numeric Extension** - Create a different output file with a numeric extension for each batch exported. Each time data is exported to this device, the number in the file extension increments by 1 or to the next available number. The name of the output file in the **Path Name** field must contain a numeric extension, which begins with any number from 000-999. For example, if you specified a 000 extension in the **Path Name** field and you export four batches, they will have extensions 000, 001, 002, and 003. If you delete the output file with a 002 extension and then export two more batches to this device, they will have extensions 002 and 004.

13. In the **List Options** area, select the option that determines when the batches listed in the BLF file are exported as a group. This is for use with the JobFlow software only. The two options are considered mutually exclusive; only one of them should be set for a single export session. Options are:

    - o **Count** - Specifies the number of batches to group together. When this number of batches sent to the export queue is reached, the queue is processed.

    - o **Time** - Indicates the time the export is performed once a day. The time must be entered using the 24-hour clock format HHMM. Typing 0000 in this field represents a blank field.

    The BLF file is not constantly polled by the software to determine whether the count or time has been reached. Instead, it is polled when a batch is added to the BLF. For example, if you set the **Time** option for 1400, the BLF is not exported until the next batch is added to it after 14:00. If the current time is greater than or equal to the device time, and if BLF file's creation time is greater than the device time, then the export is processed.

14. You can force an existing .BLF to export, even if the COUNT or TIME have not been reached as specified in the export device. To do this:

    a. Assuming the BLF is in the \FormWare\Files folder or a subfolder, add the following line to the [Files] section of the SendTo. ini file : `ITEM6=BLF Export,fwedtlng.exe –EXP,` where ITEM# is whatever # is next.

    b. Optionally, specify the export program and export device in the above command using the following as part of the –EXP parameter: `-EXP:<program name>,<device name>`

    c. Right-click on the BLF file in the Files folder and select the **BLF Export** option from the context menu.

    d. Delete the BLF file after the export process is complete.

15. Select **OK** to accept your changes and return to the Job Properties, Export tab.

Specifying Export Devices

You can specify an export device within a job, within the Client Parameters of a JobFlow task, or from the Batch Processing dialog within Administration.

- To assign a device to a job, specify the device in the **Export device** field on the Job Properties, Export tab.

- To assign a device to a JobFlow task, which overrides the device specified in the job, specify the device in the **Client Parameter** field of the Task Properties, General tab in the JobFlow Designer.

- To assign a device that overrides the device specified in the job during ad hoc processing, specify a device in the **Device Name** field of the Batch Processing dialog. Access this dialog from Administration by highlighting a batch or a BDF file and selecting the **Process > Batch Processing** menu option.

Setting Job Properties for an FPL Export

When conducting an FPL export session, you can export data using the properties set in the job or ad hoc. The more common method of exporting data is to set properties in a job and then conduct the FPL export process using the JobFlow FPL Export task. The -EXP JobFlow client parameter is the default for the FPL Export task. If you specify the -EXP parameter for the FPL Export task, the job export properties do not effect the export session and you must modify the task parameters.

You can also conduct ad hoc export sessions from the Administration module. Although you can still use the export properties set within a job, you can also specify an export program and device to apply to batch data files produced from a number of different jobs. The ad hoc method uses the **Batch Processing** option from the Process menu in the Administration module.

The Export property page of the Job Properties dialog in the Designer module contains options that define the behavior of the export session. The following section gives you an overview of how these options impact exporting data.

The following table details the results of conducting an FPL export process and using the export program and device specified for a job in job properties.

| | | | **Export Session Results** | | |
Batches in Export Session	JobFlow	.BLF *	When start & Release at end conditions occur**	Values of local variables reset on Job change	Files/ Indexes close, Devices reset on Job change
Same Job	No	No	Once/session	No	No
Different Jobs	No	No	Once/session	Yes	Yes
Same Job	Yes	No	Once/batch	Yes	Yes
Same Job	Yes	Yes	Once/session	No	No
Different Jobs	Yes	No	Once/batch	Yes	Yes
Different Jobs	Yes	Yes	Once/session	Yes/No	Yes/No

* Only with JobFlow

** These conditions can occur either once per export session or once per each batch of an export session

The following table details the results of conducting an FPL export process and using either the export program and device specified by JobFlow parameters or the export program and device specified in the Batch Processing dialog accessed from the Administration module. Both of these methods override the export options set in the job's properties.

Batches in Export Session	JobFlow	.BLF *	Export Session Results		
			When start & release at end conditions occur**	Values of local variables reset on Job change	Files/ Indexes close, Devices reset on Job change
Same Job	No	No	Once/session	No	No
Different Jobs	No	No	Once/session	No	No
Same Job	Yes	No	Once/batch	Yes	Yes
Same Job	Yes	Yes	Once/session	No	No
Different Jobs	Yes	No	Once/batch	Yes	Yes
Different Jobs	Yes	Yes	Once/session	No	No

* Only with JobFlow

** These conditions can occur either once per export session or once per each batch of an export session.

Setting Field Properties for an FPL Export

You can assign export properties on a field-by-field basis instead of using the job export properties. The following section gives you an overview of how field export options impact exporting data through FPL.

Before you can set the field property, you must specify the Forms export option for the FPL export. The Forms export option exports only those fields with assigned export positions (i.e., a value in the Export position field). If the Forms export option is not specified, setting an export position for fields will not impact the export of data. You can specify the Forms export option within:

- As a job property within Designer
- JobFlow Client
- Administration module

Note: If you set an **Export Position** value for one or more fields, do not set job export properties. Do not combine these two types of properties for exporting as these are different approaches to setting up a job for exporting data.

To set field properties for an FPL export:

1. Open a job in Designer.
2. Display the Job Properties, Export tab for the job.
3. In the **Defaults** area, select the **Form Defined** option.
4. Open a form and double-click on the field for which you want to set properties.
5. Display the Field Properties, Checkbox Edits tab for the field.

6. In the **Export Position** field, select the number of the character position for all fields that you need to export with this method. To avoid overwriting data, be sure to set the character position far apart enough from the field data to accommodate the entire length of the data field.

Setting the Forms Export Option from JobFlow Designer

You can set the Forms export option from the JobFlow Client rather than from within Designer. You must specify the -EXB:F parameter for the FPL Export task. When a batch is processed with the JobFlow task, the Forms export option is applied to the batch data. To do this:

1. From Administration, select the **JobFlow Designer** button. The JobFlow Designer window displays.

2. Open an existing JobFlow.

3. Select the **Export** task button. The Export Types dialog displays.

4. Select the **FormWare Programming Language** option and then select **OK**.

5. Double-click on the **Export – FormWare Programming Language** task in the JobFlow Browser to display the Properties dialog for that task.

6. In the **Client Parameters** field on the General Properties tab, type **-EXB:F,<device name>** where <device name> is the name of an existing device.

7. Select **Save** to save your changes to the JobFlow.

8. Exit JobFlow Designer and return to Administration.

Setting the Forms Export Option from Administration

Follow these steps to apply the Forms export option to ad hoc batch export processing from the Administration module:

1. From the File Browser within Administration, select the Files folder to display the batch data files. Batch data files bear the same name as the BDF but do not have a file extension. (If no files display in the Contents area, select the **Show All Files** button on the toolbar.)

2. Select one or more batches within the Contents area and then select **Batch Processing** from Process menu. The Batch Processing dialog displays.

3. In the **Processing types** area, select the **Export** option to enable the options that relate to export processing.

4. In the **Programming Name** field in the **Processing Options** area, type the export program name.

5. In the **Device Name** field, type the name of an existing device.

6. In the **Export Options** area, select the **Forms** option.

7. Select **OK** to export the batches you selected using the specified export parameters.

Running the FPL Export

Creating a Task in JobFlow Designer

To run the FPL export process in a JobFlow, you can either use the predefined Export task for FPL or create a Custom task within JobFlow and use the –EXB parameter. You can override the

job export properties for either type of JobFlow task within JobFlow Designer by specifying an export program or export device. To do this:

1. From Administration, select the **JobFlow Designer** toolbar button. The JobFlow Designer window displays.

2. Open an existing JobFlow.

3. Select the **Export** task button. The Export Types dialog displays.

4. Select the **FormWare Programming Language** option and then select **OK**.

5. Double-click on the **Export – FormWare Programming Language** task in the JobFlow Browser to display the Properties dialog for that task.

6. In the **Client Parameters** field on the General Properties tab, type the parameters that you want to modify the behavior of an FPL export task. The following client parameters can be used to specify how to format the exported data for an FPL export process.

Keyword	Parameter Code	Definition
Export, comma delimited	-EXB:C,\<device\>	Exports all data in a comma-delimited format. This parameter corresponds to the Comma Delimited export option. The device name is not optional, it must be specified for this parameter.
Export, by position	-EXB:F,\<device\>	Executes an export according to the export positions set in Designer on a field-by-field basis. No other formatting is done. The device name is not optional, it must be specified for this parameter. The export position can be found on the Field Properties, Checkbox Edits tab.
		This parameter corresponds to the Forms export option in Administration.
Export, form defined	-EXB:G,\<device\>	Exports data in a text file format. The device name is not optional, it must be specified for this parameter. Unlike other formats, this format is not universal. Data is output to a text file with the description=value format, with the following syntax:
		Image = ImagePath
		form = FormName
		fieldname= FieldData
		You can retrieve this file in a text editor to determine a means of extracting the data into another source.
		The following is a sample data file exported with the –EXB:G export format:
		IMAGENAME=c:\formware\images\ DEMO0005.TIF
		FORMNAME=DOCFRONT
		PrimName=DONALD MILLER
		SSN=464645645
		address=6345 ANYSTREET ROAD
		CITY=SCRANTON
		STATE=PA
		ZIP=18501

Keyword	Parameter Code	Definition
Export, no formatting	-EXB:N, <device>	Exports data from record 1, field 1 to the end of the batch, working sequentially through the batch without any formatting. The device name is not optional, but must be specified for this parameter. This parameter corresponds to the Data Records export option in Administration.
Export	-EXP:<program>, <device>	Executes an export program. If no program or device is specified in the parameter, information is taken from the job.

7. Select **Save** to save your changes to the JobFlow.

8. Exit JobFlow Designer and return to Administration.

Using the FPL Export Task with Export Protections

When using the predefined FPL Export task, any failed export protections return a code of 28. You can use the predefined FPL Export task and route return code 28 to another JobFlow task. This option provides the operator with an Export Protection Violation message for any batch that fails the chosen export protections.

Select **OK** on this message dialog to process the batch as normal and continue exporting with the remaining batches. Select **Cancel** to prompt the operator N more times (where N = number of retries) with the Export Protection Violation message. Then, give a JobFlow message of that the batch failed processing and has been routed to the task defined for return code 28.

Select **OK** in the second error message to continue exporting the remaining batches. Select **Cancel** to abort the Export task. All skipped batches are routed to the task defined for return code 28.

Running FPL Export from the Command Line

You can conduct an FPL export process from the command line of a DOS prompt. The following sections provide information on how to do this.

To export data using the job properties from the command line, the simplest syntax is:

```
FWExpLng -EXP c:\formware\files\*
```

This command runs the export programs specified in the jobs that were used to create the batches. The pathname listed is for the batches you want to export.

You can also specify a name of an export program that will be used on all of the batches in this export session instead of the export program specified in their corresponding jobs. The command syntax is:

```
FWExpLng -EXP:<program name>,<device> c:\formware\files\*
```

In this command, <program name> is the name of the export program, and <device> is the name of the device used for export.

Using the Custom Export Task with Export Protections

When conducting an FPL export process, any failed export protections will return a code of 28. You can use a JobFlow Custom task and define the return code of 28 for export. For Client Parameters, enter the following:

```
-EXP:<program name>,<device> -EPE=101
```

Where <program name> and <device> are optional. The -EPE=101 parameter routes to the JobFlow task defined in return code 101 for any Export Protection Error.

Add the client return code of 101 and route it to the desired task. This option will automatically route any batch that fails the chosen export protections to the task defined for return code 101, then continue exporting with the remaining batches.

Using Show Statements to Run FPL Export

During an FPL export process, you can display show statements in the Export Progress dialog. A show statement is static text that appears in the progress dialog. It is not dynamic text, which would change its display in the dialog during the export process. To do this, add a switch to either the Client Program Name option for the JobFlow task, or to the command line to display the show statements. The syntax is:

```
FWExpLng -EDT –EXP:<program name>,<device>
```

In this command, the -EDT switch turns on the show statements. Without this switch and the show statement in the export program, the Export Progress dialog will only contain a progress bar. The export program and device names are optional.

Running FPL Export from Administration

Batch processing executes standalone FPL programs against one or more batches. To do this:

1. Open Administration.

2. From the Files folder, select a batch or batches.

3. Select **Batch Processing** from the **Process** menu to display the Batch Processing dialog.

4. In the **Processing types** area, select the program type to execute against the batch from the following options. The fields and options activate/deactivate based on the option you select in this area.
    - o **Edit** - Executes a file edit program against one or more batches. The **Edit** option requires you to fill in only the **Program Name** field.
    - o **Export** - Executes an export program against one or more batches. A valid export device must also be specified in the **Device Name** field.
    - o **Sort** - Executes a sort program against one or more batches. If you select this option, you must also specify an **Index Name**.

5. To use the export program specified in the job, leave the **Program Name** field blank. If you type a new name, it must be a compiled (.r) program.

6. To use the export device specified in the job, leave the **Device Name** field blank.

7. In the **Index Name** field, type the index name created if a sort program is used.

8. In the **Key Size** field, select the size of the key field if you are using a sort program.

9. In the **Export Options** area, select from the following options:
    - o **Forms** - Exports the batch data with the export position and record size specified in the FDF.
    - o **Data Records** - Exports all the data in each record—data without formatting. If the export device is a print device, each record ends with a carriage return and a line feed.

o   **Comma Delimited** - Exports all of the data in each record, separating each field's data with a comma. If the export device is a print device, each record ends with a carriage return and a line feed.

10. Select **OK** to run the export process.

Configuring an OTG AX Export

FormWare tightly integrates with Online Technologies Group, Inc.'s (OTG) Application Xtender (AX) system resulting in a simple and effective document capture, storage, and retrieval product. The FormWare software provides document scanning, image enhancement, image and data quality assurance, and data capture. The integrated product enables users of the combined system to easily access stored documents in the OTG AX document management system.

OTG AX export is the process by which images and/or index information from FormWare is transferred to the OTG AX document management system for storage. After data is successfully exported using the OTG AX export component, the images are available for query and retrieval through the OTG AX document management system.

Before you create a FormWare job for export to an OTG AX document management system, finalize the design of both the FormWare job and the corresponding OTG AX document management system. This may take some planning time to ensure that the fields in both the FormWare job and the OTG AX document management system conform to the same data type, length, and other considerations. Be cautious when assigning properties to the fields that could impede the export process. FormWare does not check during the design phase to verify that the OTG AX document management system properties are compatible with those in the FormWare job. Conflicts may only become apparent at the time of export, such as exporting alphabetic data to a numeric-only OTG AX field.

After you have created a FormWare job, data form, fields, and JobFlow task for an OTG AX export, you must configure the FormWare system for that export. To allow FormWare to export to an OTG AX document management system, you must select an OTG AX application at the job level and configure the OTG AX Export component at the form level.

Setting Job Properties for OTG AX Export

Perform the following steps to select an OTG AX application at the job level:

1. Open a job in Designer.

2. Display the Job Properties, General tab for the job.

3. In the **Component** field, select the **OTG AX Export** option.

4. Select the **Configure** button to display the OTG AX Application dialog for the job.

5. In the **Application Name** field, select a new application. The corresponding description for the application displays in the **Application Description** field.

6. Select the **OK** button to accept your changes and return to Designer.

Setting Form Properties for OTG AX Export

Perform the following steps to configure the OTG AX component at the form level within Designer:

1. Open a job and form in Designer.

2. Display the Form Properties, General tab for the form.

3. In the **Component** field, select the **OTG AX Export** option.

4. Select the **Configure** button to display the OTG AX Configuration dialog. The **Non-Index Fields** list contains the complete list of available FormWare field names within this job. The **Index Fields** list contains a list of the FormWare field names with the corresponding OTG AX Index field name in parentheses.

5. Highlight a field name in the **Non-Index Fields** list and select the **Add** button. The OTG AX Field Mapping dialog displays.

6. In the **AX field name** field, select a OTG AX export mapping name. By default, the FormWare field name displays in this field unless there is a corresponding AX field name. When you select a mapping name, the matching FormWare field name displays in the **FormWare field name** field.

7. Select **OK** to return to the OTG AX Configuration dialog. The FormWare field name and the assigned OTG AX field name moves to the **Index Fields** list box.

8. To remove a field from the **Index Fields** list, highlight the name in the **Index Fields** list and select the **Remove** button. The field name moves back to the **Non-Index Fields** list.

9. To add or remove multiple fields to and from the **Index Fields** list, select the fields while holding down the [Shift] or [Ctrl] key. Then select the **Add** or **Remove** button. When you select multiple fields to add, the FormWare field names are used as OTG AX field names.

10. To modify the name of any field in the **Index Fields** list, double-click on the field name to display the OTG AX Field Mapping dialog. Select a new mapping name from the **AX field name** field and select **OK**.

Note: If the OTG AX field name used in FormWare does not match an existing OTG AX profile field name in the OTG AX document management system, a new profile field is created at the time of export. The new profile field name in the OTG AX document management system will be the OTG AX field name used in FormWare. The data type of the new profile field will be text.

Setting Clean Up Properties

Setting Clean Up Properties in Designer

The last step in your forms processing application is Clean Up. This step removes files that are no longer necessary to save hardware space. To establish Clean Up properties for a job:

1. Open the job in Designer.

2. Display the Job Properties, Cleanup tab for the job.

3. Select the file types you want to delete from the following:
    o   **Batch Files** - Deletes all batch data files created by the current job processes.

- o **BDF Files** - Deletes all BDF files used by the current job processes.
- o **Image Files** - Deletes all image files using the current job.
- o **Enhanced Image Files** - Deletes only enhanced image files using the current job.
- o **Thumbnail Image Files** - Deletes only thumbnail images files using the current job.
- o **Color Image Files** - Deletes only color images files using the current job.
- o **OCR Files** - Deletes all files using the current job.
- o **Multi-page TIFF Files** - Deletes an image file determined to be a multi-page TIFF file and the corresponding split TIFF files. The split TIFF files have the same base name as the original image, but with incrementing numeric extensions. The original multi-page TIFF file has that base name with a .tif extension. Therefore, if you select this option, and TIFF files with these naming schemes are found, the multi-page TIFF and the corresponding split TIFF files are deleted.
- o **Sub-directory** - Deletes all sub-directory folders using the current job. Only one layer of sub-directories is deleted.
- o **I-Form Transaction Files** - Deletes all I-Form transaction files associated with a JobFlow BDF.
- o **Virtual Form ID Data Files** – Deletes all data files associated with virtual Form ID.

4. Select the **Save All** button to save your settings for the job.

Establishing the JobFlow

JobFlow is the FormWare workflow software designed specifically for data capture applications. JobFlow is easy to use, yet sophisticated enough to manage the most difficult application. The intuitive interface of the JobFlow Designer module allows you to create a workflow system in a few minutes. Even complex functions, such as error handling and batch routing, can easily be accomplished with a click of the mouse.

JobFlow works with all of the FormWare processing modules. Without the JobFlow system, processing data would require direct supervision, and the chance for failure would be high. The JobFlow module tracks images through the scanning, Image Pre-Processing (IPP), and Recognition steps. It then assigns batches to operators for Completion. JobFlow also routes batches through the verification process and finally through the export process. This processing of batches through all tasks provides easy management of data capture applications.

Understanding the JobFlow Process

Before you create a JobFlow system, you must first determine which functions are needed to process your data. These processing tasks are directly dependent upon the type of data on the form, the expected volume of forms processed each day, and the time it takes to perform each task on a batch of forms. For example, if you need to process 100,000 paper documents per day and the pre-processing time needed for each image allows you to process only 75,000 per day, you would need two IPP workstations.

The following diagram represents a typical JobFlow configuration. The arrows indicate the flow of images through a JobFlow system.

JobFlow automates processing by routing data from task to task without human intervention. These unattended JobFlow tasks perform many of the repetitive chores associated with forms

processing. This means a reduction in the number of personnel required, and shifting existing staff to responsibilities that require more human intervention.

In a forms processing application that processes low volumes, you can use a single workstation to perform multiple tasks. Images are routed through each task, but all processing is performed on the same workstation. This only occurs if the task switching functionality is enabled. For example, both the Form ID and Image Enhancement tasks can be run within your processing time constraints on the same workstation.

Understanding JobFlow Tasks

You can add the following tasks to your JobFlow:

- **Image Pre-Processing (IPP)** - The IPP JobFlow task cleans up scanned (or faxed) images to prepare them for Form ID and Recognition.

- **Image Quality Check (IQC)** - The IQC JobFlow task verifies the quality of images either automatically or manually. IQC can address such image problems as incorrect orientation, holes, damaged pages, or stretched images.
  - *Automatic IQC* surveys and rejects images based on thresholds specified for the job in the Designer module. A BDF with a rejected image can then be routed to other processes, such as IPP or Manual IQC, to address the problem.
  - *Manual IQC* displays thumbnails of the images in a BDF so an operator can reject or rescan the images.

- **Form Identification (Form ID)** - The Form ID JobFlow task associates a data form with each image. The identified data form and its properties are then applied to the image during all subsequent processes in the JobFlow.

- **Enhancement** - The Enhancement JobFlow task further enhances a pre-processed image by removing lines and other elements from the image to prepare it for Recognition.

- **Recognition** - The Recognition JobFlow task executes the InScript, Nestor, AEG, Mitek, CGK, Parascript, or Mark Sense recognition engine. Recognition engines interpret patterns on images and convert them into data characters. Recognition engines can also process rules to conduct data validation checks. The Bar Code JobFlow task executes the Visionshape bar code recognition engine, which performs recognition on bar codes on the image.

- **Edits** - The Edits JobFlow task executes an edit program against the batch. These are standalone edits that are distinct from the job properties and can be run against a BDF, OCR file, or batch data file. You can write the following types of edit programs:
  - *Pre-Completion Edits* – Use the FormWare Programming Language (FPL) to execute a standalone edit before running a Completion session.
  - *Post-Completion* Edits – Use FPL to execute a standalone edit after running a Completion session.
  - *InScript Rules* – Use the Rules Language to execute an edit program on data forms and fields. You can execute this edit type during post-Recognition or post-Completion.

Note: InScript Rules can be run within a FormWare application that uses any of the available recognition engines.

  - *Visual Basic* – Use a VBA edit project to execute edit routines. Specify the VBA project in the client parameters for the task using the following syntax: `-VBP=<projectname>.vbp`

- **Batch Build** - The Batch Build JobFlow task converts the Recognition results files (OCR files) into batch data files. It also builds an index to all low confidence characters in the batch, which improves the speed of Reject Repair. You must add this task to your JobFlow if you are performing batch-based Reject Repair .

- **Completion** - The Completion JobFlow task runs Completion sessions where Completion operators type and verify data. You must select the mode of Completion. By creating a separate task for each mode, you can use all or several of them in your JobFlow. Select from the following modes:

  o *New* - Starts a new batch for reject repair or data entry from image or paper (key from image).

  o *Append* - Opens an existing batch and enables operators to process new records at the end of the batch.

  o *Verify* - Opens the batch so an operator can verify the data from recognition and reject repair. You set the verification option for each field in Designer.

  o *Correct* - Opens the batch so an operator can fix data. To change any data, the operator must press a correction key.

  o *Update* - Opens the batch so that the cursor only stops on fields that contain the update indicator.

  o *Remove Flags* - Opens the batch and the program searches for error flags (inserted by previous operators or programmatically) to repair. When the program finds a flag, it positions the cursor in the first character position of the field.

  o *Quality Check* - Opens a batch for quality check where an operator verifies every n^{th} record. You must set the QCR parameter in an INI file to designate which records to present to the quality check operator.

  o *Read Only* - Opens the batch in read only mode. The operator can press [Enter] or the [Down Arrow] to move from field to field; however, no changes can be made to field data.

  o *Add New Images* - Appends new images to an existing batch. If you add new images, you must place the new image names after all previously-processed images in the BDF. The program compares the image list stored with the batch to the image list in the BDF. If it finds a discrepancy, it adds the new images and positions the cursor on the first appended image record for processing.

  o *Identify Images* - Processes the images that failed Form ID. Each image that failed is presented for identification; the operator identifies the form's number and moves to the next image. The BDF is updated with the form number and the batch is routed to Recognition or Reject Repair.

  o *Manual Training* - Manually train a new form in Completion

  o *Batch Build* - Takes a list of images in a BDF and their associated OCR data and creates a batch data file.

Completion also allows you to create multi-pass modes to create a division of work for operators based on field type. The following modes are available:

  o *None* - Multi-pass is not used. This is the default setting.

  o *Numeric Only* - Presents only numeric data fields in this pass.

  o Alpha *Only* - Pre*sents only alphabetic data fields in this pass.*

  o *Alpha* Numeric Only - Presents alphanumeric data fields, such as Any Type fields in this pass.

- o *Non-Numeric Only* - Presents alphanumeric data fields, such as an Any Type recognition field in this pass.
- o *Update Only* - Presents data fields that have the Update option enabled in this pass.
- o *Verify Only* - Presents data fields that have the Verify option enabled in this pass.
- o *Specified* - Presents data fields that were assigned the corresponding Display Mode numeric value specified here. You assign data fields a numeric display mode value in the **Display Mode** field on the Field Properties, General tab in Designer.
- o *Display* - Displays groups of fields to an operator that are unrelated to a distinct field type. For example, if you want to group fields for an address label, assign those fields to a specific Display Mode. When that Display Mode number is referenced in JobFlow, then only those fields are shown to the operator.

- **Export**- The Export JobFlow task allows you to export data in a batch data file to an outside source, such as a text file or document management system. You can export batch data to the following destinations.
  - o *Visual Basic* - Exports batch data to a Visual Basic program. This allows the user to create and use their own Visual Basic program to store exported data.
  - o *ODBC* - Exports batch data to a specified ODBC-compliant database, database table, and database columns.
  - o *XML* - Exports batch data to an XML-formatted, flat text file. The XML export component generates one output file per transaction or batch.
  - o *FPL* - Converts batch data into formatted ASCII text using FPL.
  - o *FileNet* - Exports batch data to a FileNet repository.
  - o *Unisys InfoImage* - Exports batch data to a specified InfoImage RBE repository.
  - o *Optika eMedia* - Exports both index data and images from FormWare into the Optika eMedia document management system. Documents submitted to Optika using this process can be archived and retrieved at a later point.
  - o *IMR Alchemy* - Exports batch data to a specified IMR Alchemy repository.
  - o *OTG Application Extender* - Exports batch data to a specified OTG AX document management system.

- **Cleanup** - The Cleanup JobFlow task deletes elements associated with processing a batch. The clean up process deletes the batch data file, BDF, images, and OCR files from the hard drive where they are stored. You determine the file types to delete in Designer.

- **Custom Task** - Using a custom task enables you to add a customized program to your JobFlow. It also lets you modify how a FormWare program or predefined tasks works in your JobFlow. For example, you can run only the Bound Client in a workflow by specifying it in a Custom JobFlow, because a predefined JobFlow task is not provided for the Bound Client. When you use a custom task, you specify the program name and any parameters on the Task Properties, General tab. A Custom task is the only JobFlow task in which you can create user-defined Client return Codes.

Establishing Edit Workstations

The ability to use edit workstations is an important part of the JobFlow system. Edit workstations are attached to the network and poll the JobFlow queue files looking for work to process. When the edit workstation finds a batch to process, the edit program is run against the data. Edit programs can perform table lookups, data calculations, check-digit calculations, and other mathematical functions on the data. Edit programs are designed to run as unattended processes, allowing operators and supervisory personnel to concentrate on other tasks.

Another feature of edit workstations is the ability to control the flow of work. If the edit program encounters an error during the execution against a batch, it can flag the error and reroute the work to the Remove Flags mode of Completion. If the program finds no errors, it can add the batch to the export queue. Allowing program control over the workflow is an important part of the JobFlow system.

Controlling Work Assignments

JobFlow also allows you to control how work is assigned to Completion operators. When an operator logs into the JobFlow system, the operator is granted certain access rights and permissions, and is assigned work to process based on the login information entered. The automated assignment option prevents operators from typing in the wrong batch name or accessing the wrong job. The JobFlow module assigns the work, and when it is finished, it immediately routes it to the next task.

Creating a JobFlow

Creating a JobFlow is as simple as selecting task icons from the JobFlow Task toolbar. JobFlow tasks should display in the JobFlow Browser in the routing order. Before you create a new JobFlow system, make a list on paper of the tasks you want to include in your JobFlow and their order. You can either insert or import a JobFlow task.

The JobFlow Task toolbar displays on the left side of the JobFlow Designer module window. You use this toolbar to add JobFlow tasks to the JobFlow system. The order in which the tasks are processed in a typical forms processing application is also the order that JobFlow task buttons display on this toolbar (from top to bottom).

A JobFlow task executes a FormWare module or process. When you assign a batch to a JobFlow task, the module the task calls then processes the images in that batch. You specify the modules and task-specific properties applied to a batch data file or BDF in JobFlow Designer; however, these properties are dependent upon the job properties specified in the BDF using Designer.

Creating a New JobFlow

To create a new JobFlow:

1. From Administration, select the **JobFlow Designer** toolbar button. The JobFlow Designer window displays.

2. Select the **New** button on the JobFlow Designer toolbar. The New JobFlow dialog displays. (To open a JobFlow, select the **Open** button, highlight the JobFlow on the Open dialog and select **Open**.)

3. Type a name for your JobFlow and select **OK**.

4. You can modify the JobFlow Designer window with the functions on the View menu. A checkmark appears to the left of an option when it is selected. Options include:
   - **Toolbar** - Displays the toolbar at the top of the JobFlow Designer window.
   - **Status Bar** - Displays the status bar at the bottom of the JobFlow Designer window.
   - **Split** - Activates the divider pane between the JobFlow Browser and the rest of the JobFlow Designer window. Click and drag the divider pane to resize how the areas on the window are partitioned.

5. Double-click on the JobFlow name in the JobFlow Browser to display the JobFlow Properties dialog.

6. Select the JobFlow Task toolbar buttons in the order you want the tasks to run. The tasks display in the JobFlow Browser in sequential order. If you need to add a task that is not represented by a toolbar button, select the Custom Task button. Depending upon the task you select, you may need to make additional choices. For example, when you select the **Completion** task button, the Completion Modes dialog displays where you select the Processing Mode and Multi-Pass mode for the Completion task you are defining.

Setting General Properties for a Task

1. Double-click on an existing JobFlow task in the JobFlow Browser to display the Task Properties dialog.

2. To anchor the display of Task Properties dialog, select the **Push Pin**. The Task Properties dialog remains visible as you select different JobFlow tasks from the JobFlow Browser. Otherwise, the Task Properties dialog closes when you select a different task.

3. In the **Client Program Name** field on the Task Properties, General tab, accept or type the program that will run when this JobFlow task is called. You must type a program name if you are defining a Custom task.

4. You can specify to run either the OCX or EXE version of a FormWare component. Add either the .ocx or .exe extension to the program name in the **Client Program Name** field. If you do not specify an extension, the EXE version is used. You can configure your JobFlow to run a combination of EXE and OCX programs.

5. OCX versions of FormWare programs do not close between transactions; instead, they remain in memory until the JobFlow Client closes. Using the OCX version is more efficient because it does not take time to close and then reopen when work becomes available. This is important when conducting transaction processing or when batch sizes are small. When you run an OCX version of a program, it remains open and waits for BDFs to be routed to it to process them. However, there are overall memory considerations when using OCX versions, since all of the OCX programs in a JobFlow system are open in memory.

6. The EXE version of a program opens and closes each time a batch is processed. This can be useful if the workstation on which the program is running has limited processing memory.

Note: The **Task Number** field displays the task number for the current task. Task numbers are assigned based on task creation order, and cannot be modified. The task number displays as the task file extension in the JobFlow folder in Administration. For example, the task file for task number 15 in the Test JobFlow is named Test.015.

---

7. In the **Client Parameters** field, accept or type any client parameters or switches you want to control the client program. Many FormWare components have client parameters or additional switches that cause the component to behave in a specific manner. See Client Parameters in the Supplementary Information section for a list of parameters.

8. In the **Task Description** field, accept or type the text used to describe this JobFlow task. This text displays in the JobFlow Browser as well as any other reference to the task, such as when an operator is logging on to a JobFlow. Customizing the task description can be useful in managing JobFlow systems with similar tasks.

9. In the **Icon Name** field, accept or type the filename for the graphic file that displays for the JobFlow task in the JobFlow Browser and in the JobFlow Monitor. The icon must be a BMP file located in the \FormWare\Programs folder.

10. Select **Save** on the JobFlow Designer window toolbar to save your changes.

Setting Transaction Routing Properties for a Task

Transaction routing defines what happens when the current task has finished or has an error.

1. Within JobFlow Designer, display the Task Properties dialog for an existing task in a JobFlow.

2. Display the Transaction Routing tab to define the routing for the current task.

3. Select the **Forward Immediately** check box to forward transactions to the next JobFlow task. This is for tasks that do not require the BDF for processing.

4. Select the **Final Task** check box if this is the last task in the JobFlow. You must select this field for the last task in a JobFlow when you use task switching.

5. In the **Route to Task** field, select the next task to which transactions will be routed. Only tasks in the current JobFlow display in the list. The default value is the next task in the order you selected tasks for the JobFlow. If you selected the JobFlow tasks in the order they will be run in the JobFlow, then you do not have to modify this value.

6. In the **Batch Failure Percentage** field, select the percentage of images in a batch that must fail the process before batch failure routing is invoked.

7. In the **Transaction Failure Routing** field, select the task to which failed transactions will be routed. Only tasks within the current JobFlow display in the list.

8. Select **Save** on the JobFlow Designer window toolbar to save your changes.

Setting Transaction Processing Properties for a Task

Transaction processing controls the flow of transaction files through the JobFlow. For some forms processing applications, all images in a transaction must be processed before the transaction can move forward to the next task. Transaction processing allows you to further customize routing.

1. Within JobFlow Designer, display the Task Properties dialog for an existing task in a JobFlow.

2. Display the Transaction Processing tab.

3. Select the **Allow Operators to Reprocess Their Own Work** check box if you want operators to reprocess work they have seen previously. Deselect this check box to prevent operators from verifying transactions they keyed.

Note: This option is available only for Completion tasks and is controlled by the operator's user name.

4. Select the **Allow Operators to Select Transactions** check box to let operators select a transaction to process at the task level when working in JobFlow Client. When this option is not selected, operators cannot select transactions to process at the task level, but must process the transactions in the order they are received.

Note: This option is available for Completion Export and Custom tasks.

5. In the **Image Processing Time** field, select the expected processing time of an image for this task in seconds. If the transaction takes longer than this amount of time, it fails after the Retry Count.

6. In the **Transaction Retry Count** field, select the number of times the JobFlow Client will attempt to process a transaction that has failed this task.

7. Select the **Allow Hardcopy Processing** check box to allow paper transactions to be processed. The Hard Copy options allow you to specify that information is going to be keyed from a paper document rather than from an image or using Recognition. If you do not select this option, the paper transactions processed by the JobFlow are ignored.

8. In the **Expected Hardcopy Batch Size** field, select the maximum batch size allowed in this task, if you are processing paper transactions in this JobFlow. This value calculates the time the JobFlow Client is active before a timeout occurs; it does not limit the number of records in a batch. Always set this value to a number higher than the expected batch size.

9. Select **Save** on the JobFlow Designer window toolbar to save your changes.

Setting Task Control Properties for a Task

Task Control properties define the task type and whether it will switch to another task.

1. Within JobFlow Designer, display the Task Properties dialog for an existing task in a JobFlow.

2. Display the Task Control tab.

3. In the **Task Type** area, select the value that determines the type of task being performed. Values are:
   - o **Operator Only** - Requires operator input. A server task does not require operator input and can run unattended. You must switch to an operator function from this task.
   - o **Server Only** - Allows only server functions to be switched to from this task.
   - o **Server or Operator** - Allows the workstation to switch between operator and server functions.
   - o **Non-Redirectable** - Prevents task switching on the workstation.

4. Select the **Enable Task Switching** check box to activate the options in the **Task Switching Method** area. These options determine the criteria for switching from running one task to another without having to log off the JobFlow Client. Task switching reduces the number of workstations required for processing. It also prevents a workstation from becoming idle. As soon as a workstation no longer has transactions to process for a task, it switches to another task to find available transactions to process. This switching to find work method keeps batch throughput high.

5. In the **Task Switching Method** area, select from the following:
   - o **Fixed Processing** - The JobFlow Client processes the number of transactions specified by the Maximum Transactions parameter before it switches JobFlow tasks. If the queue becomes empty before the maximum is reached, then a task switch occurs when the number of transactions equals zero. This approach helps prevent processing bottlenecks. In the **Maximum Transactions** field, select the number of transactions to process before JobFlow Client switches tasks.
   - o **Load Balancing** - The JobFlow Client examines all queues for the number of transactions pending. The task that has the highest number of transactions pending becomes the active task. Load balancing attempts to keep the number of available transactions in each queue balanced. Switching takes place based on the number of transactions pending – not the maximum number of transactions or an empty queue.

6. In the **Next JobFlow** field in the **Destination** area, accept the current JobFlow or select a new one to switch to when the specified criteria are met.

7. In the **Next Task** field, select the next task to perform in the JobFlow selected.

8. Select **Save** on the JobFlow Designer window toolbar to save your changes.

Setting Client Control Properties for a Task

Client Control properties specify how many times a transaction is retried before it fails the task. Other options allow you to control wait times.

1. Within JobFlow Designer, display the Task Properties dialog for an existing task in a JobFlow.

2. Select the Client Control tab.

3. In the **Client Control Retry Count** field, select the number of times to retry the transaction instead of failing if a network problem occurs. The default is three.

4. In the **Client Retry Pause** field, select the number of seconds to pause the JobFlow Client before transaction processing is retried.

5. In the **Out of Work Pause** field, select the number of seconds to pause the JobFlow Client and display a message when the task has no transactions to process.

6. In the **Error Message Response Pause** field, select the number of seconds to pause the JobFlow Client and display an error message when the task encounters a processing problem.

7. Select the **Run Minimized** check box to run the default message boxes for this task in "minimized" mode. You can select the dialog from the task bar to display the message. You can also select the Minimize button on a message dialog to minimize, rather than close, it.

8. When you select the **Run Minimized** check box, it turns off the Error Beep function for error dialogs. If you want to hear the beep for tasks that have the **Run Minimized** function turned on, you must add the following INI file parameter to the appropriate System.ini or job INI file:

    ```
    FWERRORBEEP=X,Y,Z
    ```

    where:

    - o   X is the frequency of the beep tone in Hz (Range 37 - 32767, suggested value 22050).
    - o   Y is the duration of the beep in milliseconds (must be greater than zero, suggested value 500).
    - o   Z is the how often the beep should occur in milliseconds (must be greater than Y, suggested value 2000).

9. In the **Success Pause** field, select the number of seconds to pause the JobFlow Client and display a message when the task function is successful. The JobFlow Client continues processing when the time specified expires.

10. Select **Save** on the JobFlow Designer window toolbar to save your changes.

Setting Client Return Code Properties for a Task

Client return codes route transactions based on the codes returned by the JobFlow Client. The Properties, Client Return Codes tab displays a list of client return codes for the current JobFlow task. You can define client return codes for the Custom task, Completion, or Export.

By default, if a processing problem occurs with a transaction, the transaction does not move forward in the JobFlow queue. It stays in the existing queue for the task that encountered the

problem. However, the transaction that caused the problem is marked as "In Process" when you examine the JobFlow in the JobFlow Monitor. This can be confusing because there are more transactions that are "In Process" than are assigned to machines or operators.

We strongly recommend that you create a reject queue to handle problem transactions and that you modify the client return codes so that any problem transactions are forwarded to this queue. If you use a reject queue, transactions that fail a particular process are not left in the original queue in an "In Process" state. Rather, they are forwarded to the reject queue. A supervisor can then determine the cause of failure and manually route the transactions to the next task.

1. Within JobFlow Designer, display the Task Properties dialog for an existing task in a JobFlow.

2. Select the Client Return Codes tab. This tab displays the following information.

    o **Return** - Displays the client return code number.

    o **Route To** - Displays the JobFlow task that the transaction is routed to if the client return code is encountered.

    o **Allow Repeats** - Allows the JobFlow to send transactions back to the component issuing the return code to process again if a client return code is returned. This is done the number of times specified in the **Transaction Retry Count** field on the Transaction Processing tab.

    o **Treat As Error** - Determines whether to treat this client return code as an error when it is encountered. If the value is Yes, an error message is issued and the retry property is applied. If the value is No, an error message is not displayed and retry processing is not attempted.

    o **Error Message** - Displays the syntax of the error message when the error code is encountered.

3. To modify a client return code, double-click a return code in the **Return** column. The Client Return Information dialog displays.

Note: Not all client return codes can be modified. Some codes are reserved by the program to perform specific functions and cannot be changed.

---

4. Select the **Allow Transaction Repeat** check box to enable the JobFlow to send transactions back to the component issuing the return code to process again if a client return code is returned. This is done the number of times specified in the **Transaction Retry Count** field on the Transaction Processing tab.

5. Select the **Treat As An Error** check box to indicate whether to treat this client return code as an error when it is encountered. If this box is selected, an error message is issued and the retry property is applied. If this box is not selected, an error message is not displayed and retry processing is not attempted.

6. In the **Return Code** field, type the numeric value for the client return code. You can use up to 100 client return codes per task. Return codes 1 to 99 are reserved by the system for internal use.

7. In the **Forward To** field, select the task designated as the reject queue.

8. In the **Error Message** field, type any changes to the problem description.

9. Select **OK** to accept your changes and return to the Client Return Codes tab.

10. Select **Save** on the JobFlow Designer window to save your changes.

Adding a Client Return Code

Several tasks in the JobFlow Client allow you to control transaction routing. The system reserves client return codes 1-100. When you create a client return code, you can use client return code numbers 100-32,767. You can assign up to 100 client return codes per task. Each task has a varying amount of system-supplied client return codes to which you can add more codes, up to 100 total.

See the FormWare Programming Reference Guide for additional information on specifying client return codes in a VBA or FPL program.

To create a new client return code for a Custom, Completion, or an Export Task:

1. Create a new task in a JobFlow to which you want to add one or more client return codes.

2. Display the Task Properties, Client Return Codes tab for the task.

3. Select the **Add** button to create a new return code. The Client Return Information dialog displays.

4. Select the **Allow Transaction Repeat** check box to allow the JobFlow to send transactions back to the component issuing the return code to process again if a client return code is returned. This is done the number of times specified in the **Transaction Retry Count** field on the Transaction Processing tab

5. Select the **Treat As An Error** check box to issue an error message and apply the retry property. If you do not select this check box, an error message is not displayed and retry processing is not attempted.

6. In the **Return Code** field, type the numeric value for the client return code. You can use up to 100 client return codes per task. Return codes 1 to 99 are reserved by the system for internal use.

7. In the **Forward To** field, select the task to which the transaction should be routed when this code is returned.

8. In the **Error Message** field, type a description of the problem. This field is enabled if you select the **Treat As An Error** check box.

9. Select **OK** to accept your changes and return to the Client Return Codes tab.

10. Select **Save** on the JobFlow Designer window to save your changes.

Creating a Batch Build Task

Completion – New mode is used to create a batch data file from a group of images and their associated OCR data. During the Completion – New task, the operator is also presented with low confidence characters and rejected fields for correction. When no low confidence characters or rejected fields exist for a batch, Completion – New mode will open the batch and then quickly close it again. Although no correction is being made to the data, the program does create a batch data file in this situation.

It is also possible to create a batch data file using a Completion – Batch Build task. Batch Build takes a list of images in a BDF and their associated OCR data and creates a batch data file. Using a command line parameter with the Batch Build task, it is possible to return a custom return code that will route that batch past the Completion – New task so the batch will not be presented to an operator for keying when no error characters exist. To accomplish this in a JobFlow, you must insert a Completion – Batch Build task before the Completion – New task.

The main function of the Batch Build program is to pre-build FormWare batch files. Optionally, the reject repair window can also be pre-built so that the rejects for all of the images in the batch are shown at once vs. just the rejects for the current image. If the Batch Build program does not detect any rejects, it can be configured to send back a unique return code, which the workflow can use to bypass the reject repair step in the workflow.

To insert a Completion – Batch Build task in a JobFlow:

1. Open JobFlow Designer.

2. Create a new JobFlow with a Completion – Batch Build task before a Completion – New task.

3. Double-click on the Batch Build task to display the Task Properties dialog.

4. Display the Client Return Codes tab for this task.

5. Select the **Add** button to add a new client return code. The Client Return Information dialog displays.

6. Create a new client return code of 100 that routes the batch past the Completion – New task to a subsequent task (e.g., Completion – Correct).

7. Select **OK** to accept this new code.

8. Save your changes to the JobFlow.

Creating a Completion Manual Training Task

You can use the ManTrain.vbp VBA project within JobFlow to manually train a new form in Completion. When this VBP is defined as a Manual Training task, operators can key form ID fields, manually zone form data fields, and enable FormWare to train that form.

Note: Manual Training using this method can only be performed on images within a production environment that have undergone the IPP and image rotation. It is essential to use images created on the actual production scanner, with the same resolution, image placement, and borders that will be used during production.

To define a Manual Training task in JobFlow Designer:

1. Open a JobFlow in JobFlow Designer.

2. Select the **Completion** button on the JobFlow task toolbar. The Completion Modes dialog displays.

3. In the **Processing Mode** area, select **New**.

4. In the **Multi-Pass Mode** area, select **None**.

5. Double-click the task to display the Properties, General tab.

6. In the **Client Parameters** field, type the following command:

    ```
    -e -VBP=ManTrain
    ```

7. In the **Task Description** field, type: Manual Training

8. Select **Save** on the JobFlow Designer window to save your changes and exit JobFlow Designer.

Using VBA with the Remove Flags Mode of Completion

In the Completion - Remove Flags mode there are a series of functions that run prior to processing an image. One of these functions is an optimization to see if there are any flagged fields. If there are not, then the field functions are disabled. The NEP=N INI parameter calls OnFormStart before these functions are run and before all data is available. When you use the NEP=Y parameter all of the data is available.

In Remove Flags mode there are not any flags and the field functions have been disabled. Therefore, the field flags are recounted after the OnFormStart condition, which allows you to flag fields in OnFormStart and have them processed. This feature allows a second pass of Completion with Remove Flags and AutoTrain Verification to run reliably.

Creating an Export Task

Once export properties are set, you can use any of the predefined Export tasks or create a Custom task. The Custom task allows you to use custom client return codes. When exporting with JobFlow, the order of the exported data is the order in which batches are sent to the Export task.

The ODBC export process always uses the properties configured in the job. You do not need to specify an export program or device for the ODBC export process.

Adding a Predefined Export Task

The predefined Export JobFlow task is an essential piece in the JobFlow system. Exporting data captured by FormWare is typically the final step in forms processing, prior to Clean Up. To add a predefined export task to your JobFlow:

1. Open a JobFlow in JobFlow Designer.

2. Select the **Export** task button on the JobFlow Task toolbar. The Export Types dialog displays.

3. Select the desired export option and select the **OK** button. The predefined Export task displays in the JobFlow Browser with the name of your selection.

Modifying a Predefined Export Task

To modify an Export task (e.g., to enable task switching, override the job export properties, assign additional parameters, etc.):

1. In JobFlow Designer, double-click on the Export task in the JobFlow Browser to display the Task Properties dialog.

2. The program name of the export component displays in the **Client Program Name** field on the General tab. For example, if you are modifying an FPL export task, FWExpLng displays in this field.

Note: It is not advisable to change the program name on this dialog. If you want to run another export component, create a new Export task for a different export component type.

3. In the **Client Parameters** field, type any new client parameters for this export task. See the Client Parameters section of Supplementary Information for details on the available parameters

4. In the **Task Description** field, type the name of the export task.

5. In the **Icon Name** field, type the file name of the icon to display for this task. This icon must be a BMP file and reside in the \FormWare\Programs folder.

6. Continue to the other tabs in the Task Properties dialog to modify the export task.

7. Select the **Save** button on the JobFlow Designer window to save your changes.

Creating a Custom Export Task

The Custom JobFlow task calls a FormWare program and then creates customized client return codes for that program. You can create a Custom JobFlow task and specify an export program and any client parameters you might need.

To add a client return code to an Export task, you must create a Custom JobFlow task. The default Export JobFlow task has a set of default client return codes and you cannot add return codes to it.

To create a custom export task:

1. Open JobFlow Designer.

2. Open an existing JobFlow or create a new one.

3. Select the **Custom** button on the JobFlow Task toolbar.

4. Double-click on the Custom task in the JobFlow Browser to display the Properties dialog.

5. In the **Client Program Name** field, type the name of the program to run. This name must be a valid EXE program and reside in the \FormWare\Programs directory unless you include a qualified path to the program.

6. In the **Client Parameters** field, type the program arguments to run with the program. See the Client Parameters section in Supplementary Information for more information.

7. In the **Task Description** field, type the descriptive name of this task.

8. Select the Client Return Codes tab.

9. Select the **Add** button to display the Client Return Information dialog.

10. Type the appropriate return code values and the select **OK**. See the Adding a Client Return Code section for more information.

11. Select **Save** on the JobFlow Designer window to save your changes.

Modifying JobFlow Tasks

Using the JobFlow Browser context menu, you can rename or delete tasks from a JobFlow. To do this:

1. To rename the JobFlow, right-click on the JobFlow name and select the **Rename** option. In the Rename JobFlow dialog, type a new name and select **OK**. You can also access the Rename JobFlow dialog by right-clicking on any task and selecting the **Rename** option; however, this option only renames the JobFlow not the selected task.

2. To remove a task from the JobFlow, right-click on the task name in the JobFlow Browser and select the **Delete** option.

3. Select **Yes** on the confirmation dialog.

Configuring the Bound Client

The Bound Client enables you to configure or *bind* processes to run as a single task. Multiple tasks can then run in an apparently seamless fashion.

Adding a Bound Client

To add a Bound Client:

1. From the Administration window, select **Tools > Configure > Bound Client**. The Bound Client Configuration dialog displays.

2. In the **Configuration** field, select or type the name of the Bound Client configuration you are defining. Use only alphanumeric and underscore characters in the name.

3. Select one or more components in the **Add Component** field by selecting a component name and then clicking **Add**. The component displays in the **Components** list.

4. The components list displays each component or process in the order that it will be executed within the Bound Client. To change the order of execution, highlight a component name and select either the **Move Up** or **Move Down** button.

Note: Carefully examine all contingent relationships between the processes you are binding and order them appropriately within the components list. Some components depend on others to run successfully. For example, if your Bound Client includes manual Image Quality Check (IQC), you must first run Image Pre-Processing (IPP) to generate the thumbnail images that IQC uses.

5. To remove a component from the list, highlight the component and select **Remove**.

6. Select **OK** to save your changes and return to the Administration window.

Setting Bound Client Properties

Two properties support the Bound Client component: SAVETEMPFILES and EXCLUDE_FILES.

Use the SAVETEMPFILES property to retain the locally copied files after they have been processed and returned to the server. This property is only used for debugging purposes since the retained files would then have to be cleaned up manually. Use SAVETEMPFILES=Y to use this property. The default is N (No).

Use the EXCLUDE_FILES property to exclude certain file types from being copied locally to achieve maximum transmission efficiency. This property uses a value similar to the CLEANUP property where each character represents a different image type, as follows:

- I = Bi-tonal images
- E = Enhanced images
- T = Thumbnail images
- C = Color images
- O = OCR data files
- V = FPO data files

Creating a Bound Client Custom JobFlow Task

To run the Bound Client as part of a JobFlow, you must create a Custom JobFlow task. To create this task:

1. Open a JobFlow or create a new one in JobFlow Designer.

2. Highlight the task in the JobFlow Browser immediately preceding the JobFlow location where you want to insert the Custom task.

3. Select the **Custom Task** button on the JobFlow Task toolbar.

4. Double-click on the Custom task in the JobFlow Browser to display the Task Properties dialog.

5. In the **Client Program Name** field, type BNDCLNT.EXE to run the Bound Client program.

6. In the **Client Parameters** field, type –BNDCFG=<configname>, where <configname> is the name you defined for this Bound Client in the **Configuration** field of the Bound Client Configuration dialog.

7. In the **Task Description** field, type a meaningful description of the bound processes. This description will display in the JobFlow Browser for this task.

8. Select **Save** to save your changes to the JobFlow.

Importing and Exporting JobFlow Tasks and Properties

You can import and export JobFlow tasks or JobFlow properties from and to other JobFlows.

Importing a JobFlow Task

When importing a task, only the properties written into the JobFlow Definition File (JFD) are imported. This process does not affect how batches in the current JobFlow task queue are processed. When you import JobFlow properties, the registration number descriptions, JobFlow Monitor view settings, and log batch activity settings are imported. To import a task from one JobFlow into another:

1. Open a JobFlow in JobFlow Designer.

2. Right-click on the JobFlow task in the JobFlow Browser to display the context menu. The import process will *replace* this task with the one you are importing.

3. Select the **Import** option. The Import dialog displays the JobFlow tasks in the \FormWare\JobFlow folder by JobFlow name and task number.

4. Highlight a JobFlow task and select the **Open** button. The name and icon for the imported task display in place of the original JobFlow task in the JobFlow Browser. The selected JobFlow task changes to reflect the JobFlow properties of the imported task, but maintains the task number of the original task.

5. Select **Save** on the JobFlow Designer window to save your changes.

Importing JobFlow Properties

Perform the following steps to import JobFlow properties from one JobFlow to another:

1. Open a JobFlow in JobFlow Designer into which you want to import JobFlow properties.

2. Right-click on the JobFlow name in the JobFlow Browser to display the context menu.

3. Select **Import**. The Open Files dialog displays showing the JobFlow files in the \FormWare\JobFlow folder.

4. Highlight a JobFlow name and select the **Open** button. The name and icon for the JobFlow do not change.

5. Select **Save** on the JobFlow Designer window to save your changes.

Exporting a JobFlow Task

When exporting a task, only the properties written in the JobFlow Definition File (JFD) are exported. This process does not affect how batches in the current JobFlow task queue are processed. When you export JobFlow properties, the registration number descriptions, JobFlow Monitor view settings, and log batch activity settings are exported. The export task process creates a new task in the JobFlow system to which you are exporting. To export a task from one JobFlow to another JobFlow:

1. Open a JobFlow in JobFlow Designer from which you want to export a task.

2. Right-click on the task in the JobFlow Browser to display the context menu.

3. Select the **Export** option. The Export dialog displays the JobFlow task files in the \FormWare\JobFlow folder by JobFlow name and task number.

4. In the **File name** field, type the name of the new JobFlow task. Be sure to include the correct JobFlow name and three-character numeric extension as part of the new file name or an error will display. This number does not need to match the original task number.

5. Select **Save** on the JobFlow Designer window to save the new task to the \FormWare\JobFlow folder. The exported JobFlow task contains the JobFlow properties from the original JobFlow task.

Exporting JobFlow Properties

To export JobFlow properties from one JobFlow to another:

1. Open a JobFlow in JobFlow Designer from which you want to export JobFlow-level properties.

2. Right-click on the JobFlow name in the JobFlow Browser to display the context menu.

3. Select the **Export** option. The Export Files dialog displays the JobFlow files in the \FormWare\JobFlow folder.

4. In the **File name** field, type the name of the new JobFlow to create for the export.

5. Select **Save** on the JobFlow Designer window. Exporting a JobFlow creates a new file.

Establishing Batch Order in the JobFlow

Using Registration Numbers

Registration numbers allow supervisory personnel to monitor groups of batches in the JobFlow. You can assign up to eight registration numbers to a given batch. You can assign registration numbers with values from 1 to 1,000 in the Scan Designer module.

For example, registration number one can represent the year. Registration number two can represent the Julian date. Registration number three can represent a number associated with the job. One or all registration numbers can be specified for monitoring. This allows supervisors to quickly examine all or part of the work currently in production. Registration numbers display in JobFlow Maintenance and are written into the batch header.

In addition to the number, you can also create a detailed description for each registration number. Perform the following steps to create the registration number descriptions.

1. Open a JobFlow in JobFlow Designer.

2. Double-click on the JobFlow name in the JobFlow Browser to display the JobFlow Properties dialog.

3. Select any value in the **Number** column to display the Registration Description dialog.

4. Type a description in the field on the dialog and select **OK**.

    The new description displays in the Job Properties dialog in the Description column for that registration number.

5. Select **Save** on the JobFlow Designer window to accept your changes.

Once you create descriptions for the registration numbers you can assign them to Scan Configuration (SCG) files that are created using the Scan Designer module. You can also add registration numbers to BDF files within the Administration module.

Creating Batch-Based Reject Repair Applications

By default, when you create a Reject Repair application, you are creating a *record-based* application. This means the low-confidence characters for each record are presented in the ticker tape. Batch-based reject repair enables Completion operators to view *all* rejected characters for the batch in the reject ticker tape, rather than just the rejects for the current record.

If there are only a few rejects per form, you can have all of the low-confidence characters for the entire batch displayed in the ticker tape. Viewing all of the low-confidence characters in a batch enables the Completion operator to look ahead and perform faster reject repair.

Batch-based reject repair functions only with batches that have less than 1,000 recognition rejects. If a batch has more than 1,000, the software reverts to record-based reject repair.

To perform batch-based reject repair, you must create a batch prior to performing reject repair. A batch build converts the Recognition results files into a FormWare batch data file. This function also builds a table of all the low-confidence characters so batch-based repair can be performed.

To create a batch-based reject repair application:

1. Use the Designer module to create a job that uses Recognition technology (i.e., has recognition zones on the setup image).

2. On the Designer window, stretch the reject ticker tape area so that multiple lines of the ticker tape are visible.

3. Display the Job Properties, Recognition tab for the job.

4. Select the **Batch Reject Repair** check box to enable batch-based reject repair.

5. Select **Save All** on the Designer window to save your changes.

Creating a Multi-Pass Application

The Captiva Software approach to forms processing is to divide the work between many workstations and enable data entry operators to maintain high throughput. This division of work also includes the reject repair process. The repair of low confidence characters can be divided among operators by field type or some other user-defined criteria, so that an operator only views and repairs characters of a specified field type.

For example, if the repair process is divided by field type, the first-pass operator may repair only alphabetic fields, the second-pass operator may repair only numeric fields, and the last-pass operator may repair only alphanumeric fields. By dividing the repair process among three operators, each operator is free to concentrate on one type of data. This increases the processing throughput.

You can design a multi-pass application by field type or display mode. In the field type method, the fields are presented to each operator based on the type of field. With this method, create your application as usual and then use the JobFlow Designer module to specify different multi-pass modes of Completion. The display mode method gives you control over which labels and fields display to the operator in each pass.

Creating a Multi-Pass Application Using Field Type

To create a multi-pass JobFlow application within Completion passes:

1. Open a JobFlow in JobFlow Designer.

2. Select the **Completion** button from the JobFlow task toolbar to add a Completion task to your JobFlow. The Completion Modes dialog displays.

3. In the **Processing Mode** area, select **New** to create a Completion – New task.

4. In the **Multi-Pass Mode** area, select the type of multi-pass application you want to create.

5. Repeat the preceding steps to create additional passes in the JobFlow. You can add as many passes as you wish – each new pass is a separate task in the JobFlow.

6. Select **Save** on the JobFlow Designer window to save your changes to the JobFlow before you exit JobFlow Designer.

You can use the multi-pass option with any of the Completion modes, except for the Identify Images mode.

Creating a Multi-Pass Application Using Display Mode

In addition to adding passes by field type, you can also design a multi-pass application by adding a user-defined field parameter to each field in the job. This user-defined parameter is called a Display Mode and is set up within the Designer module.

Note: To function correctly, every form must have at least one field for each pass. This field must have a related label and cannot be a non-display field. However, the field can be an auto-skip field. For example, if you are using three passes, Form 1 must have two auto-skip fields, one set for pass 2 and the other for pass 3. Form 2 must have two auto-skip fields, one for pass 1 and the other for pass 3. Form 3 must have two auto-skip fields, one for pass 1 and one for pass 2.

To create a multi-pass application based on display mode:

1. Open a job in Designer.

2. Double-click on the field you want to use in the Job Browser and display the Field Properties, General tab for the field.

3. In the **Display Mode** field, select the number of the pass within the multi-pass application to which this field is assigned.

4. Repeat the preceding steps as necessary for any additional fields on the data form.

5. Select **Save** on the Designer window to save your changes before you exit the Designer module.

6. Open the JobFlow with these fields in JobFlow Designer.

7. Select the **Completion** button in the JobFlow task toolbar to add a Completion task to your JobFlow. The Completion Modes dialog displays.

8. In the **Processing Mode** area, select the **New** option to create a Completion – New task.

9. In the **Multi-Pass Mode** area, select the **Specified** option. This actives the next field.

10. Select the number you assigned in the **Display Mode** field to one or more fields (on the Field Properties, General tab) for this job in Designer.

11. Repeat the preceding steps as necessary to create a Completion task for each display mode you created for this job.

12. Select **Save** on the JobFlow Designer window to save your changes to the job before exiting.

Configuring the Split and Join Utility

You can use the FormWare Split and Join utility (ISJClnt) to divide a multi-page TIFF file into individual TIFF files, or build a single, multi-page TIFF file from a number of separate TIFF files. You create a Custom JobFlow task to use ISJClnt. You must split a multi-page TIFF into individual image files before any FormWare module can process it; therefore, you must call the Split utility prior to any FormWare processing. You can call the Join utility at any time during processing.

Both the Split and Join utility functions are called using the ISJClnt program. Use the –ISJMODE=SPLIT parameter to run the Split utility and the –ISJMODE=JOIN parameter to run the Join utility. By default, the Join utility uses the images in the BDF to build the new multi-page image.

Important Note: Additional information after the file name in the BDF is not allowed in either split or joint mode. For example, the following line in a BDF is not allowed: "c:\formware\images\img00001.tif ADDITIONAL_INFO".

Note: Before running the Split utility, you must set the **File extension** field on the Job Properties, Recognition tab to "???". If you leave the value as "OCR", the recognition files will be overwritten. The BDF must have the OCRDATA=<path> parameter where <path> is the directory where the OCR files should be placed. In FormWare File System, you can add this parameter to the BDF by inserting a comment during scanning.

Using the Split Utility

You can have any number of multi-page TIFFs in a single BDF. In the BDF, the image requires a .TIF extension. Following is a sample BDF:

```
JOBNAME=IB40
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.TIF
```

When you run the Split utility against the BDF, the multi-page image is processed, breaking it into individual TIFF files. The single TIFF files have the same base file name as the original TIFF file, but are assigned numeric extensions, beginning with .000 and incrementing by one until .999 is reached. This allows 1,000 individual images to be split from a single multi-page TIFF. The individual TIFF files are created in the same folder as the original multi-page TIFF. The BDF is then updated to reflect the names and paths of these new image files.

To create a job containing separate TIFF files from the images in a multi-page BDF file:

1. Open a JobFlow in JobFlow Designer.

2. Select the **Custom Task** button on the JobFlow task toolbar to create a custom task.

3. Double-click on the new Custom task within the JobFlow Browser to display the Properties, General tab .

4. In the **Client Program Name** field, type: ISJClnt. This calls the Split and Join Utility program.

5. In the **Client Parameters** field, type -ISJMODE=SPLIT. This parameter indicates that you want to call the Split function within the Split and Join Utility. The original multi-page TIFF image remains intact and in the same location after the Split utility processes it.

    This is how the BDF in the example above looks after the Split utility processes it:

```
JOBNAME=IB40
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.000
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.001
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.002
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.003
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.004
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.005
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.006
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.007
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.008
```

6. Select the **Save** button to save your changes.

Using the Join Utility

The Join utility builds a single, multi-page TIFF of the images listed in a BDF. Use this utility when images have been reordered, deleted, or inserted during the Completion process or when you want to store the images after the IPP or Image Enhancement processes have improved the quality of the images.

By default, the Join utility uses the images in the BDF. These images do not require a numeric extension or the same base file name. The following is a sample BDF:

```
JOBNAME=IB40
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.001
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.002
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.006
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.005
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.004
C:\FORMWARE\IMAGES\SPLIT\MULTIPG.003
```

The Join utility takes the base name from the first image in the BDF, adds a .tif extension, and then combines all the images into a single, multi-page TIFF. When creating a multi-page TIFF, you must be careful that there is not an existing image with the same name as the multi-page image being created.

To create a multi-page TIFF from the images in a batch data file:

1. Open the JobFlow in JobFlow Designer.

2. Select the **Custom Task** button on the JobFlow task toolbar to create a custom task within your JobFlow.

3. Double-click on the new task in the JobFlow Browser to display the Properties, General tab.

4. In the **Client Program Name** field, type: ISJClnt. This calls the Split and Join Utility program.

5. In the **Client Parameters** field, type -ISJMODE=JOIN.

6. Select **Save** on the JobFlow Designer window to save your JobFlow settings.

Testing the JobFlow

While you are developing an application in the Designer module, you are encouraged to test field, form, and job properties at different levels of the application before you implement the application in a production environment. You can also create and test a JobFlow from Designer to test applications within a JobFlow working environment.

Creating a Test JobFlow in Designer

Follow these steps to create a JobFlow in the Designer module while you are testing job properties.

1. Open a job in the Designer module.

2. Display the Job Properties, Test tab.

3. Select the check boxes for the processes you want in the JobFlow.

4. Select the **Generate JobFlow** button. The Enter Name dialog displays.

5. In the **Name** field, type the name of the JobFlow you want to create and then select the **OK** button. This displays a JobFlow created from the processes selected on this tab.

6. Modify the JobFlow as necessary before proceeding with the test. For example, set any additional properties for the JobFlow, such as enabling task switching within the Task Properties dialog.

7. Select **Save** on the JobFlow Designer window to save your changes.

8. **Exit** the JobFlow Designer window and return to the Designer module.

Testing a JobFlow in Designer

You can test a JobFlow from the Designer module as you fine-tune your FormWare application. The JobFlow test is independent of the current job in the Designer module. You can run a number of different BDFs through the JobFlow test. You can run the JobFlow test on an existing JobFlow, or you can create and test a JobFlow within Designer.

To test a JobFlow in Designer:

1. Open a job in Designer.

2. Display the Job Properties, Test tab for the job.

3. Select the **Test JobFlow** button. This displays the Browse Files dialog.

4. Highlight an existing JobFlow and select the **Open** button. The Test JobFlow dialog displays.

5. In the **Batch Definition File** field, select the **Browse** button to select a BDF to use in the JobFlow test. The job properties for this job are used in the JobFlow test. This can be a job other than the one currently open in Designer module. Highlight the BDF and select **OK**.

6. On the Test JobFlow dialog, select the **Edit** button to display the BDF in a text editor where you can modify it. Make any changes, then save and close the dialog.

7. Select the **Check JobFlow** button to report any errors for job properties that are not supported in the JobFlow. For example, an error is returned if a registration field exists in the job but a registration task does not exist in the JobFlow. This option reports multiple errors, if they exist.

8. Select the **Select** button for the Available Task field. The Available Task dialog displays the list of JobFlow tasks in the current JobFlow. **Double-click** on a task to select it for the test.

9. Select the **Single Step** check box to run a single JobFlow task at a time and review the results of each. If you do not select this option, the JobFlow task is run and then each of the subsequent tasks in the JobFlow are run.

10. Select the **Start** button to initiate the JobFlow test. A progress dialog displays for the current task. When the task is complete, information regarding the test of the task displays in the **Current Task** area of the Test JobFlow dialog. While the test is running, this button displays as the **Stop** button.

11. The **Current Task** area displays the following results:

    o **Description** - Description of the current JobFlow task.

    o **Program** - Program name for the current JobFlow task.

    o **Command Line** - Command line parameter that you specified for the current task as well as the path to the BDF you selected for testing.

    o **Result** - Client return code for the current JobFlow task.

o   **Error Message** - Error message for the client return code.

12. Select the **Next Task** button to run the JobFlow task displayed. This runs the next task in the JobFlow according to the routing properties you specified. This button is disabled if the current task generates an error.

13. Continue to run tests in the Test JobFlow dialog to fine-tune both the JobFlow and the job for your production environment.

14. When you are finished testing, select the **OK** button to return to the Designer module.

Understanding Distributed Processing

The globalization of many organizations requires an efficient forms processing solution for their distributed workforce. The FormWare Distributed Processing components provide the ability to share the workload among different physical locations not connected by a Local Area Network (LAN).

For example, an insurance company chooses to outsource a portion of their forms processing. Branch offices scan claim forms and other documents in-house and send those images over the Internet to a third-party facility for processing. A portion of that processing may involve the use of off-site data entry operators. Those operators also use the Internet to download batches of work from the facility, and send the processed data back for further work.

The *host site* is the location where your Web Server, Internet FormWare Server (IFS), and FormWare Server reside. These servers contain data and programs that other computers can access by means of a network or modem. Workstations reside at *remote sites* that are some distance from your host site. A remote site is accessible from your host site by cables or other communications links.

Remote input clients and *remote processing clients* perform distributed processing in the FormWare system.

- **Remote input clients** create new work that enters a host site JobFlow for continued processing. The remote input clients, Distributed Scan, FormWare Filler, I-Forms, and XML Import, all push work into an existing JobFlow.

- **Remote processing clients** connect to an existing JobFlow at the host site. They receive transaction data from the host site, process it, and return the modified data to the host site for further processing.

The *remote input clients* include:

- **Distributed Scan** is the process by which remote clients configured with scanners run the FormWare Scan Operator to create images and Batch Definition files (BDFs), which are then sent to a FormWare JobFlow via the Internet.

- **FormWare Filler** is a data entry application that enables a remote user to operate in a simplified Completion New or Completion Append mode, creating new batches of data to submit to a JobFlow on the FormWare Server for continued processing.

- **XML Import** provides a way to import work from XML-formatted data from a remote location into a FormWare JobFlow.

- **I-Forms** provides a method for users to submit data captured on HTML forms over the Internet. The completed I-Form enters the FormWare JobFlow, which further extracts and processes the data.

The *remote processing clients* include:

- **Distributed Completion** connects the remote workstation to a FormWare Server at a host site over the Internet. It also connects a remote client to a host JobFlow and enables a Completion operator outside of the host site to work on that portion of the JobFlow. In a Distributed Completion environment, Completion operators connect their workstation to a FormWare Server via the Internet at a separate host site and download batches to their remote workstation for processing.

- **Distributed WorkFlow** connects two FormWare Servers, each with their own separate JobFlow, over the Internet. It processes data at one host site and then transfers that data to a FormWare Server at another host site for continued processing. The work can be completed at the second host site, or returned to the original host site for further processing. In a distributed environment, Completion operators retrieve their work from a JobFlow on the FormWare Server over a LAN connection at their host site.

You use the following programs and utilities to set up Distributed Processing and work with the remote components:

- **Remote Transfer Utility** – You install the Remote Transfer Utility on a remote workstation to communicate with the IFS and the FormWare Server at the host site. It enables quick communication with the server to send and receive data.

- **Remote Sync** - Remote Sync synchronizes system files on the remote client workstation with system files on the server by updating the files on the client workstation.

- **Internet FormWare Server (IFS)**- The IFS resides on a server at the host site. As the liaison between the Web server and the FormWare server, all communication between remote clients and the server is routed over the Internet, via the HTTP protocol, through a Web server to the IFS. During IFS installation, you enter the initial IFS configuration information.

- **Proxy Server** – You can use a proxy server in your distributed environment to implement security via a secure socket. The proxy server filters and passes information from the remote workstation to the Web server. The transfer of data is seamless, and all current functionality remains the same except that the end user must know the name and/or IP address of the proxy server and the port number. Although this information is recorded in each user's INI file, it does not automatically populate the connection dialog. When connecting to the host site, the remote user needs to select the **Use Proxy Server** check box in the connection dialog. This expands the dialog and displays the proxy server address field where the user types a proxy server name or an IP address, and port field where they type a port number.

Distributed Processing uses files in the following \FormWare folders:

- \FormWare\Remote – This folder contains the following sub folders:
  - \Configs – This folder contains Remote Configuration files (CFG) that identify the job being processed by the remote user, JobFlow/WorkFlow information pertaining to the remote session, and the files contained in a completed remote session transmission back to the FormWare Server. CFG files are created on the FormWare Server and then downloaded to the remote site using Remote Sync when the remote user logs on to begin processing work.
  - \WorkList – This folder contains the remote worklist files (<username>.rwl) that store information concerning batches or data items checked out to a remote user. The worklist files are generated by the remote processing component during a Distributed Completion session and can be viewed, but not modified, using the Remote Worklist Viewer (FWRWLVU.exe).

Installing and Configuring Components for Distributed Processing

Now that you understand the different components related to Distributed Processing, you are ready to install and configure a FormWare Distributed Processing application.

See the following sections within Installing & Configuring for more information.

- Installing Components for Distributed Processing – Instructions on how to install components for the three servers on the host site—the Internet FormWare Server (IFS), the FormWare File Server, and the Web Server—necessary to run Distributed Processing. Also provides instructions on Installing Remote Workstations and Installing Distributed Processing Patches.

- Configuring the FormWare File Server – Instructions on Understanding the Remote Sync utility, Creating User INI Files, and Using the Operator Statistics File.

- Configuring the Internet FormWare Server (IFS) – Instructions on configuring the IFS, which acts as a liaison between the Web Server and the FormWare File Server on the host site.

- Using Remote Scripting – Information on the methods and event handlers within a remote script used to validate and authenticate remote users and data.

- Connecting a Remote Client to a File Server – Instructions on how to successfully link a remote client to the FormWare Server via the Internet.

- Using Desktop Shortcuts to Launch File Server Components – Tips on how to create desktop shortcuts to run FormWare Server components from a remote workstation.

Processing Under SSL

FormWare will function under SSL (secure socket layer 128-bit encryption). Note the following:

- The URL used by the remote user to log in must include the "Common Name" in the certificate for FormWare to access it. The common name in the certificate should be the actual name of the web server. The net bios name OR the IP address of the web server can be used as the Common Name in a certificate.

- The URL used by the remote user to log in must be in the following syntax: https://<Certificate Common Name>/cgi-bin/fwcgi.dll. This syntax must also serve as the parameter for the RMT_HOSTNAME parameter in the user INI file.

Example #1

If the Common Name in the certificate is the net bios name of the web server, and the net bios name of the web server is: Webserver - the URL, then RMT_HOSTNAME parameter in the user INI file would be: https://Webserver/cgi-bin/fwcgi.dll. The remote user must also enter this exact URL in the Web Server Name field of the Remote Sync Connection dialog.

Example #2

If the Common Name in the certificate is the IP address of the web server, and the IP address of the web server is: 10.3.17.256, then the URL in the RMT_HOSTNAME parameter in the user INI file would be: https://10.3.17.256/cgi-bin/fwcgi.dll. The remote user must also enter this exact URL in the Web Server Name field of the Remote Sync Connection dialog.

Creating an I-Form

Internet Form (I-Form) processing provides a method for users to extend data capture to web-based forms. Once the remote user submits a form, the Web Server checks the data for completeness and accuracy. The completed I-Form enters the JobFlow where data is extracted and further processed.

Remote processing clients receive transaction data from the FormWare Server, process it, and return the modified data to the JobFlow on the server for further processing.

Internet Forms (I-Forms) processing provides a simple and seamless way for users to extend data capture to Web-based forms. As a thin client implementation, I-Forms does not require any special installation for the remote user. Any user with Internet access and a supported browser can access a Web site, make a request for a specific form, and begin entering data that can later be submitted into a FormWare JobFlow.

Via a browser, remote users make requests to access a form type. These requests are sent from the browser to a Web Server. The Web Server then communicates with an available IFS. The IFS communicates directly with the FormWare Server, locates the requested form and then generates a DHTML version of the form to display to the user.

Validations can be made within the browser to provide instant feedback and once the remote user submits a form, more sophisticated edits can be run on the server. The final step is for the I-Forms transaction to be added to a particular WorkFlow at a particular step for continued processing. You specify the WorkFlow information in a remote configuration file (*.CFG) in the \FormWare\Remote\Configs folder. The remote configuration file is associated with a job within Designer on the Job Properties, I-Forms tab.

If you want to incorporate a FormWare-generated I-Form into an existing Web site (created by a third-party HTML editor), specify the HTML file name. This file must reside in a location that is accessible to IFS (i.e., in the FormWare environment). Within the HTML file, you need the following line at the appropriate location to indicate the point of insertion for the FormWare form.:

```
<!--FORMWARE INSERT POINT-->
```

Elements of I-Forms Design

Use FormWare Designer to develop a form for users to access via the Internet. Designer supports several data objects common to Web-based forms, such as list boxes, radio buttons, and check boxes. Within Designer, you can use a setup image to help you create an I-Form, or you can create a form by placing elements on the design grid. If you use the latter method, you should have some type of layout in mind before starting the design process.

With I-Forms you can use the different Designer elements, such as buttons and drop-down menus for your forms, but you should keep in mind that current browsers render HTML documents somewhat differently. This means you should create the forms as small as possible so that you do not lose information that is too wide for the browser.

Note: You can specify HTML generation using Cascading Style Sheets (CSS) or HTML tables. Only Netscape 6.x and Internet Explorer correctly support CSS. However, due to inconsistent browser support for CSS pages, we recommend that you use HTML tables to define your form.

The following example is a portion of an I-Form used to manage online training registrations at Captiva Software Corporation.

Designing good electronic forms has certain of the same elements as paper forms, such as providing instructions for completing the form. However, references to pages, dropout ink, typing, photocopying, etc. are not relevant for I-Forms.

Follow these guidelines as you create your I-Form:

- If you are building a form from scratch, spend time planning the data fields the form should contain.
- If you are building a form from a paper model, evaluate how you should modify it to take advantage of electronic features and functionality.
- In planning the form layout, consider space issues such as maximum field length for each field, sequence of fields on the form, and groupings.
- Design field labels to clearly indicate the type of information you want to capture in each field.
- Determine the field types appropriate for the types of data on the form (e.g., text or number fields, combo boxes (drop-down menus), check boxes, and radio buttons).
- Determine which fields are required.
- Consider how to alert users of the required fields on the I-Form. For example, when the user leaves the required field or the form, display a message that reads: "This is (or Name of the field) a required field. Please enter the requested information."
- Determine how to notify users if they type invalid values into a field.

Note: When designing an application for Distributed Processing, remember that the locations of the images and BDFs change as they move from the host site to a remote workstation. Do not write edit programs that assume that this location remains the same during processing. Also, use databases or FormWare indexes for data lookup and validation that will be accessible to a remote user.

Setting I-Forms Properties

Forms within a forms processing application can be accessed over the Internet within a browser. This requires that you set I-Forms-specific properties at the job level. These properties control the completed and canceled transactions associated with the I-Form you are creating, as well as define scripting options and JobFlow setup routines.

Additional properties at the form level identify the HTML documents and images associated with the I-Form you are creating.

Setting Job-Level Properties for I-Forms

To set job-level properties for I-Forms:

1. Open a job in Designer.

2. Double-click on the job name in the Job Browser to display the Job Properties dialog.

3. Select the I-Forms tab.

4. In the **IFS Script** area, select the type of script you want to use for your job. This script will be used on the IFS for validation edits within the job. Options are: **JavaScript** or **VBScript**.

5. Select the **Disable Script** check box to temporarily disable a script to allow testing without performing edits. The script remains associated with the job, but is inactive until you deselect the **Disable Script** check box.

6. Select the **Edit** button to display the script editor for the script type you selected (i.e., **JavaScript** or **VBScript**).

7. In the **URL to Use After Canceled Transaction** field, type the URL for the web page the user is returned to if the transaction is canceled before submission. Typically, the address will point to the home page.

8. In the **URL to Use After a Suspended Transaction** field, type the URL for the web page the user is returned to if the transaction is suspended before completion. Typically, the address will point to a page that assigns a transaction ID that the operator can reference when returning to complete the transaction at a later time.

9. In the **URL to Use for Submit Action** field, type the URL for the location on the Web Server to which submitted transactions are routed. For example, if the value is https://www.captivasoftware.com/cgi-bin/fwcgi.exe, then www.captivasoftware.com is the web server name and /cgi-bin/fwcgi.exe is the name of the CGI the Web Server will load to process the post. This defaults to whichever submit action URL is designated on the System Defaults, General tab. You can append the following additional arguments to the end of the URL:

    o **$ECHO%3dy** - Echoes or repeats back information as the Web Server and the IFS attempt to connect and lets you know if a successful connection was made.

    o **$IFSHOST%3d <workstation name>** - Explicitly submits requests to the specified IFS Host workstation rather than using the first available.

    o **$FWCGIDEBUG%3ddv** - Runs in verbose debug mode. All processes return information and textual representation of the HTML form passed between the Web server and the IFS.

10. In the **HTML to Use for End of Transaction Processing** area, select the location to which the user is taken once a transaction is submitted (e.g., a confirmation page). Options are:

> o   **FormWare Generated** - The user will be taken to a FormWare-generated
>   confirmation that includes a transaction ID and other information (e.g., JobFlow
>   name).
>
> o   **FormWare Form** - The user will be taken to a form designed using a FormWare
>   FDF once a transaction is submitted. This allows you to create a confirmation page.
>   Type the name of the form in the **Form Name** field.
>
> o   **URL Only** - After submitting a transaction, the user will be taken to the URL you
>   type in the **URL** field.

11. Select the **Browse** button next to the **Remote JobFlow Configuration** field to locate the
    Remote Configuration file (CFG). Remote Configuration files are used with FormWare Filler
    (see the Defining FormWare Filler section for instructions on creating a configuration file.)
    The Browse Files dialog displays the contents of the \FormWare\Remote\Configs folder. To
    incorporate a completed I-Forms transaction into a FormWare JobFlow, you must define
    configuration settings for the JobFlow name, the task within the JobFlow where the
    completed transaction will enter, and other optional JobFlow-related information.

12. Select the **Edit** button to use the Remote Configuration Properties dialog to modify the
    selected CFG file. See the Defining a Remote Configuration File section for more
    information.

13. Select the **Clear** button to remove the selected CFG file from the **Remote JobFlow
    Configuration** field.

14. Select **Save All** on the Designer window toolbar to save your property settings for the job.

Setting Form-Level Properties for I-Forms

To set form-level properties for I-Forms:

1. Open the job in Designer.

2. Double-click on the form name in the Job Browser to display the Form Properties dialog.

3. Select the I-Forms tab.

4. Select the **Browse** button next to the **Containing HTML Document** field, to locate the
   HTML file. This is the container HTML file into which a Form Definition file (FDF) is
   inserted. The <!--FORMWARE INSERT POINT--> tag indicates the point of insertion for
   the form in the HTML file. This file must be in a location accessible to the ICAPTURE
   program (i.e., in the FormWare environment, either locally or on the FormWare Server). See
   the following sample tag:

```
<!--FORMWARE INSERT POINT-->
<form method="POST"
action="https://www.captivasoftware.com/cgi-bin/fwcgi.exe"
name="TRAINING_REGISTRATION">
```

5. In the **Submit Action for Designer Test** field, type the URL (Universal Resource Locator)
   for the location to which the Designer test submission is routed. This defaults to the web
   server address in the **Test Submit Action** field on the System Defaults, General tab. This
   allows you to direct your testing within Designer to a different location than the production
   location.

6. Select the **Disable Submit Action** check box to disable the submit action in the previous
   field. Select this option if you do not want to submit the I-Form during development (i.e., for
   debugging and testing).

7. Select the **Disable Data Commit** check box to disable transaction creation while still allowing the IFS script events to launch.

8. Select the **I-Forms Use Only** check box to disable the use of this form by Completion. This option allows you to create an I-Form that does not contain data and does not become part of the I-Forms transaction. This is useful when you use Completion to verify/correct work done via I-Forms, but Completion cannot load a form that does not have any data fields defined.

9. In the **Background Bitmap URL** field, type the URL for the location of the background image of the form.

10. Select the **Edit** button in the **Edit Script for Browser** area to display the Custom Script editor where you can create your own scripts that execute on the browser side during processing. The Custom Script Editor assumes knowledge of the scripting language and does not include a debugging utility.

11. Select the **Disable** check box in the **Edit Script for Browser** area to disable the script for debugging or testing. Although the script remains associated with the form, it is not executed as long as this option is selected.

12. Select the **Preview in Browser** button to displays the form in your system's browser (e.g., Internet Explorer, Netscape).

13. Select the **Advanced Options** button to set HTML options that determine the appearance of your form when viewed in an Internet browser. The Advanced HTML Generation Options dialog displays.

14. Select the type of layout you will use for your I-Form from the following options:
    o **Use Style Sheets for Layout** - Uses style sheets when generating an HTML document for the I-Form you are creating. See the Using Cascading Style Sheets section for more information.
    o **Use Tables for Layout** - Uses tables when generating an HTML document for the I-Form you are creating. Select from the following options to indicate placement of an object within its defined space: **Align Left, Align Center,** or **Align Right**. See the Using HTML Tables section for more information.

15. Select the **Disable Wrap** check box to disable line wrapping in the browser if you are using tables for the layout. This prevents the layout to be reformatted, which would change the way the form displays on the HTML page.

16. Select the **Disable Browser Page Cache** check box to disable the Internet browser's caching capability so that the form is loaded from the original disk every time it is accessed.

Caution: Selecting this option can cause unexpected results. When an I-Form contains a hot link (label with an associated URL), certain browsers can fail when attempting to return from the link.

---

17. In the **Unfollowed link color** field, select the color of a link that has not been selected.

18. In the **Visited link color** field, select the color of a link that has been selected.

19. In the **Activated link color** field, select the color of a link that has been selected but not released.

20. Select the **Edit** button next to the **Additional HTML for <HEAD> section** field, to display the Added <HEAD> HTML text editor where you can write custom HTML code for displaying the I-Form in a browser. For example, you can add the following keywords to the header: <META NAME="Keywords" CONTENT="MyCompany, MyProduct, MyPrice">

21. In the **Text to use for <title> tag** field, type the text for the form title in the browser. If you do not type a value the form name is used.

22. In the **Text to use for 'Must Enter' error** field, type the text for the Must Enter error when a required field is skipped. If you do not type a value, the "Must Enter Field" default is used.

23. In the **Text to use for 'Must Complete' error** field, type the text for a Must Complete error. If you do not type a value, the "Must Complete Field" default is used.

24. In the **Text to use for 'Numeric Field' error** field, type the text for a Numeric Field error when a non-numeric value is typed in a numeric field. If you do not type a value, the "Numeric Field Type" default is used.

25. In the **Text to use for 'Alpha Only' error** field, type the text for an Alpha Only error when a non-alpha character is typed in an alpha only field. If you do not type a value, the "Alpha Only Field Type" default is used.

26. In the **Text to use for 'Invalid Character' error** field, type the text for an Invalid Character error when the user types an invalid character. If you do not type a value, the "Invalid Characters Entered" default is used.

27. In the **Text to use for 'Range Check' error** field, type the text for a Range Check error when the user types a value outside the valid range. If you do not type a value, the "Range Check Error" default is used.

28. Select **OK** accept your changes and to return to the Form Properties dialog.

29. Select the **Save All** button on the Designer window toolbar to save your settings.

Setting Field-Level Properties for I-Forms

To set field-level properties for I-Forms:

1. Open the job and form in Designer.

2. Double-click on the field in the Job Browser for which you want to set I-Form properties to display the Field Properties dialog.

3. Select the I-Forms tab.

4. In the **URL** field, type the URL address you want to be assigned to this field. This can be a link to another web site or a jump to a different location within the current web site. This address indicates the location to which the user is taken after selecting the label, button, or bitmap image from within a browser.

5. In the **Bitmap** area, define the following:

    o In the **URL** field, type the URL address for the bitmap image file to be displayed in this field. You can also type a local or network server path and filename to retrieve a bitmap file.

    o In the **Height** field, type the number of pixels that specifies the height of this bitmap as it will appear in a browser.

    o In the **Width** field, type the number of pixels that specifies the width of this bitmap as it will appear in a browser.

Note: The Bitmap area changes to the Mouse Over Bitmaps area when you are defining a Bitmap Select Area field. See the Adding a Bitmap Selection Area section for more information.

6. In the **Data Source** field, select the data source for the label or input field you are creating. Select from the pre-defined script variables or type your own. The value stored within the

variable is bound to the field, and that value displays in the field during run time. For example, in a FormWare JobFlow transaction, if you want a transaction ID to be returned to a user once he has submitted an I-Form transaction, you could create a confirmation page that displays the JobFlow transaction ID.

Note: If this field is a member of an HTML table, the name of the table displays in the **Member of HTML table** field.

7. In the **Edit Script for Browser** area, define the following:
    o   Select the **Disable** check box to disable the field-level edit script for the browser
    o   Select the **At Form End** check box to enable the browser-side script to execute at the end of the form. This is the recommended setting. In most cases, this allows for the most fluid movement through the form, while still ensuring data integrity (to the extent that the browser-side edits are enabled).

8. Select from the following edit script options to define the edits that will execute against this field in the browser, if applicable. The options in the **Edit Script for Browser** area are enabled only for input, check box, radio button, list box, and combo box fields.
    o   **FormWare** - Applies FormWare-generated edits based on a subset of the field properties defined for this field. If you select FormWare, edit options set on the Checkbox Edits, Mark Sense, and Selective Edits tabs are applied. The following items are considered in the FormWare-generated browser-side field edits: Must enter; Must complete; Field Type ; Range – lower, Upper Limits (numeric fields only); Range – Include and Exclude values; Right Justify; Zero Fill If Data; and Zero Fill No Data.
    o   **Custom** - Enables a custom edit script. Select the **Edit** button to create a custom edit script in the Custom Script editor window.

9. In the Submit Type area, select the option that indicates the type of transaction submission action for this button or bitmap. Options are:
    o   **None** - Use the button for other purposes (e.g., link to another page – type the link in the **URL** field, or provide custom code for the button).
    o   **Prev** (Previous) – Moves to the previous record in the transaction. This behavior is identical to a record back in Completion. Select this if you are creating a project that will contain more than one form. This results in the $SUBMITTYPE property available in the server side script having the value of "PREV". The OnFormEnd event runs at the server.
    o   **Cancel** - Cancels the current transaction. This results in the $SUBMITTYPE property having a value of "CANCEL", the OnTransactionCancel event running at the server, and the transaction being cancelled.
    o   **Next** - Moves to the next record in the transaction if there is one. This results in the $SUBMITTYPE property having the value of "NEXT". The OnFormEnd event runs at the server.
    o   **Done** - Triggers the actual submission of the transaction to the JobFlow. This results in the $SUBMITTYPE property having a value of "DONE". The OnFormEnd event runs at the server. If your server script code does not indicate an error condition, then the OnTransactionSubmit event, followed by the OnTransactionEnd event runs at the server.
    o   **Clear** - Clears all input fields. This is handled by JavaScript in the browser.

- o **Resume** - Resumes a suspended transaction. This results in the $SUBMITTYPE property having a value of "RESUME". The OnTransactionResume event runs at the server.
- o **Suspend** - Suspends the current form's transaction. Suspend holds the transaction at the IFS and retains all work up to the point of suspension without submitting it to the JobFlow.

10. Select the **Use As Default** button to indicates that the selections in the Submit Type area will be used as the default for this and subsequent I-Forms.

11. Select the **Advanced Options** button. The Advanced HTML Generation Options dialog for the field-level displays.

12. In the **Text to use as the field name in browser error messages** field, type the text you want to appear as the field name within browser error messages for this field.

13. In the **Additional HTML for tag section** field, type any additional HTML text to include in the FormWare-generated HTML for this field's tag. For example, currently I-Forms does not generate the necessary HTML to set background colors due to browser incompatibilities. Therefore, to add HTML code to change the background color of a button to red (or any field type other than a label) type the following in this field:

    ```
    style="background=#ff0000"
    ```

14. Select the **Open URL in new browser window** check box if the field you are configuring is a bitmap, label, or button and you have attached a URL to the field. Checking this field will open the URL in a new browser window. Leaving this field unchecked will open the URL in the existing window.

15. Select **OK** to accept your changes and return to the Field Properties dialog.

16. Select **Save All** on the Designer window toolbar to save your changes.

Adding a Custom Control

You can add a custom control to an I-Form to incorporate custom Dynamic HTML (DHTML) code.

Note: Any custom control that you include, however, is outside the realm of FormWare functionality and is not validated by the Designer module.

To create a custom control area on an I-Form:

1. Open the job and form in Designer.

2. Select the **Custom Control** button on the I-Forms toolbar in the Designer window and draw the custom control area on the data form.

3. Double-click on the custom control field to display the Field Properties dialog.

4. Select the I-Forms tab.

5. In the **URL** field, type the location of the DHTML code for this field.

6. Select **Save All** on the Designer window toolbar to save your settings.

Adding a Bitmap Selection Area

When you create a Bitmap Selection Area field on top of an existing Bitmap field, you are enabling one or both of the following actions on a bitmap:

- Embedding a "hot link" on a portion of a graphic that takes the user to the indicated location (e.g., another web site or different location on the same web site).

- Changing the appearance of the selection area so that when the mouse is placed on the selection area, a different or modified graphic displays. This indicates to the user that the selection area is "active." For example, you might want to create an area on a graphic that highlights when the mouse is on the graphic. In another case, you might want to display a selection menu when the mouse is on the selection area.

To create a Bitmap Selection Area:

1. Open the job and form in Designer.

2. On the Designer window, add the bitmap field to the data form.

3. Select the **Bitmap Selection Area** toolbar button from the I-Forms toolbar on the Designer window and draw the selection area on the bitmap field on the data form.

4. Display the Field Properties, I-Forms dialog for the Bitmap Selection Area field.

5. In the **URL for On** field in the Mouse Over Bitmaps area, type the file location for the graphic to be displayed when the mouse is ON the selection area of the bitmap on the I-Form. This location is typically a URL on the Internet, but can also be a path to a network file.

6. In the **URL for Off** field, type the file location for the graphic to be displayed when the mouse is OFF the selection area of the bitmap on the I-Form. This location is typically a URL on the Internet, but can also be a path to a network file.

7. Select **Save All** on the Designer window toolbar to save your settings.

Using Cascading Style Sheets

Cascading style sheets (CSS) are a feature of HTML. A style sheet is a template describing how things should be displayed within a browser. Style sheets are referred to as cascading because more than one can be applied to each HTML page. CSS' are not fully supported by any Web browser.

The limitations of using a CSS include:

- Vertical and horizontal lines created in Designer do not display in the browser.

- The browser view is maintained as the browser window is resized; no changes in the HTML page occur. For example, if you make the browser window smaller, you will probably lose the ability to view a portion of your form.

You can use a CSS to quickly convert an existing form into an I-Form. To create an I-Form using a CSS:

1. Open Designer and create a form as you would for any FormWare job.

2. When your layout is finished, display the Form Properties dialog.

3. On the I-Forms tab, select the **Advanced Options** button. The Advanced HTML Generation Options dialog for the form-level displays.

4. Select the **Use Style Sheets for Layout** option and select **OK**. By default, Designer generates HTML using auto-generated tables. You can view the difference by previewing your web page before and after attaching a CSS.

5. On the I-Forms tab, select the **Preview in Browser** button to display your web page.

6. Return to the form in Designer to make any modifications to the form elements.

7. Select **Save All** on the Designer window toolbar to save your settings.

Adding HTML Tables

Use HTML tables when creating I-Forms to design forms with a more uniform and predictable appearance when the form is displayed in an Internet browser. HTML tables enable you to "nest" tables and other objects (fields, labels) within tables. "Nesting" is the process of inserting tables within cells of existing tables. You can create up to 20 levels of nested tables on a form. Individual cells within a table can also support custom DHTML (Dynamic Hypertext Markup Language) controls. Additionally, you can configure empty table rows to collapse at form run-time, resulting in a more concise and polished looking I-Form.

Limitations of HTML tables include:

*   Vertical lines created in Designer will display, but horizontal lines will not.
*   Tables attempt to readjust the HTML page to the current size of the browser window.

Creating an HTML Table

To create an I-Form using HTML tables:

1. Create a new job in Designer.

2. Set the Job and Form properties.

3. Define the labels and fields for your form.

4. When your layout is finished, display the Form Properties dialog.

5. On the I-Forms tab, select the **Advanced Options** button. The Advanced HTML Generation Options dialog displays.

6. Select the **Use Tables for Layout** option and the proper alignment for the table, then select **OK**.

7. Select the **Layout > HTML Tables > Create** menu option to create tables for your form elements. The HTML Table dialog displays.

Note: You may want to group certain elements on the form together and place them in separate tables. For example, group elements that require four columns separately from those elements requiring only two columns. This presents a cleaner web page as table cells can be adjusted without detrimental impact on the adjacent elements. You can also nest tables by creating a large table and placing smaller tables inside. This will allow you to create side-by-side tables.

8. In the **Name** field, accept the sequentially numbered table name or type a new table name. Spaces are not allowed in table names.

9. In the **Number of rows** field, select the number of rows that will be present in this table.

10. In the **Number of columns** field, select the number of columns that will be present in this table.

11. In the **Border** field, select the width of the table border in pixels. Setting this option to 0 indicates no border. The maximum is 4 pixels.

12. In the **Cell padding** field, select the number of pixels that determines the width of space on each side of the object in the table cell.

13. In the **Cell spacing** field, select the number of pixels that determines the width of space between table cells.

14. In the **Background color** field, select a color for the table background.

15. Select the **Transparent** check box to display the background as transparent when viewed in a browser. Anything placed on the form underneath the table will be visible. For example, you can have a bitmap graphic below a table that is visible when the form is viewed in an Internet browser.

16. In the **Border color** field, select a color for the table border. This color applies only if the **Border** field is set to a value from 1-4.

17. In the **Default Vertical Alignment** area, select the vertical alignment for the objects the cells of the table you are creating. Options are:
    o   **Top** - Objects display within the top portion of the table cells.
    o   **Middle** - Objects display within the center, or middle, portion of table cells.
    o   **Bottom** - Objects display in the bottom portion of the table cells.

18. In the **Default Horizontal Alignment** area, select the horizontal alignment for the objects in the cells of the table you are creating. Options are:
    o   **Left** - Objects display in the left portion of the table cells.
    o   **Center** - Objects display in the center portion of the table cells.
    o   **Right** - Objects display in the right portion of the table cells.

Note: When a browser renders a table-based HTML page, it will ignore ALL white space on each line, which results in the page being left aligned. One way to circumvent this action is to use HTML tables with an empty first—or left—column that starts at the left edge of the page. You can now adjust the size of this column to move the rest of the table contents to the desired location.

---

19. Select **OK** to accept your changes and return to the Designer window.

20. On the data form, use your cursor to click on the design grid to place the table. To move the table, draw a loop around the table. The table displays blue sizing handles that allow you to resize the table and table cells. When you place your cursor over the selected table, you can grab and move the table on the grid.

21. Move the form elements from the design grid into the table cells. The elements "snap" into the cell. Temporarily enlarging the cells in your table makes inserting elements easier.

22. On the Form Properties, I-Forms tab, select the **Preview in Browser** button to display your Web page.

23. Make any modifications to the tables by right-clicking on the table and selecting **HTML Table Properties** from the table context menu. Then select **Table** or **Cell.**

24. Select **Save All** on the Designer window toolbar to save your form.

Modifying HTML Table Properties

To view a list of tables for an I-Form and to access table and cell level property dialogs:

1. Open a job in Designer.

2. Select the form for which you have created HTML tables.

3. Select the **Layout > HTML Table > List** menu option. The HTML Table List dialog displays the names of the tables created for the form.

4. To modify the properties you set for this table, highlight a form name and select the **Properties** button. The HTML Table Properties dialog displays. There are additional properties on this dialog than on the Properties dialog you used to initially create the table.

5. In the **Name** field, select the name of the table or type a new one.

6. In the **Border** field, select the width of the table border in pixels. Setting this option to 0 indicates no border. The maximum setting is 4 pixels.

7. In the **Cell padding** field, select the number of pixels of space on either side of the object in the table cell.

8. In the **Cell spacing** field, select the number of pixels of space between table cells.

9. In the **Background color** field, select the table background color.

10. Select the **Transparent** check box to display the table as transparent when viewed in a browser. Anything placed on the form underneath the table will be visible. For example, you can have a bitmap graphic below a table that is visible when the form is viewed in an Internet browser.

11. In the **Border color** field, select the color of the table border.

12. In the **Upper left X position** field, select the X coordinate of the upper left corner of the table on the form.

13. In the **Upper left Y position** field, select the Y coordinate of the upper left corner of the table on the form.

14. In the **Number of rows** field, select the number of rows this table should contain.

15. In the **Number of columns** field, select the number of columns this table should contain.

16. Select the **Remove Empty Rows** check box to prevent empty table rows from displaying at run-time. To display empty rows at run-time, deselect this option.

17. Select the **Edit** button next to the **Extra HTML** field to edit any custom Dynamic HTML that has been applied to this table in the Table Extra HTML editor.

18. Select **OK** to accept your changes and return to the HTML Table List dialog.

19. To modify the number of rows and columns in the table, highlight the form name and select the **Cell** button. The HTML Table Cell Selection dialog displays. Modify the values and select **OK** to close the dialog. You cannot select numbers higher than the current number of rows or columns in the table.

20. To activate a table so that you can move or size it, highlight the table name and select the **Select** button. A dotted blue line displays around the table indicating that it is active.

21. To remove a table, highlight the table name and select the **Delete** button.

22. When you are finished, select the **Close** button.

23. Select **Save All** on the Designer window toolbar to save your changes.

Modifying HTML Cell Properties

Once you have defined a table, you can modify the properties for a specific table cell. You can apply these changes to a single cell or to all the cells in a row or column. To modify cell properties:

1. Open the form with the table in Designer.

2. Right-click on a cell in the table and select the **HTML Table Properties > Cell** option from the context menu. The HTML Cell Properties dialog displays.

3. In the **Apply Changes To** area, select one of the following options to indicate which cells in the table are affected by the changes you are making on this dialog:
   o **Current cell only**
   o **All cells of the current row**
   o **All cells of the current column**.

4. Select the **Fill to Area** check box to enable graphics within a Bitmap or Selection Area to fill the entire table cell. This applies to bitmaps, custom controls, and selection areas only.

5. In the **Vertical Alignment** area, select from the following options to determine the vertical placement of objects in the cells: **Top, Middle,** or **Bottom**.

6. In the **Horizontal Alignment** area, select from the following options to determine the horizontal placement of objects in the cells: **Left, Center,** or **Right**.

7. In the **Color** field, select the color for the background of the table cell.

8. If this HTML table cell contains a field or label from the data form, the name of that field or label displays in the **Contained field** field. This field is display only.

9. If this HTML table cell contains an imbedded HTML table, the name of that table displays in the **Contained table** field. This field is display only.

10. In the **Width** field, select the number of pixels that specifies the width of this cell's column.

11. In the **Height** field, select the number of pixels that specifies the height of this cell's row.

12. Select the **Edit** button next to the Extra HTML field to modify any custom Dynamic HTML that has been applied to this cell.

13. Select the **OK** button to return to Designer.

14. Select **Save All** on the Designer window toolbar to save your changes.

Defining Distributed XML

The Distributed XML Import component of Distributed Processing allows you to import XML-formatted data into a JobFlow for continued processing. The import component creates BDF and batch data files at the host.

Note: Using Distributed XML assumes that you are familiar with the purpose and semantics of the XML programming language.

Importing XML documents into FormWare involves two different steps: creating an XML document and importing it into FormWare.

Creating a FormWare-Compliant XML File

First you must create an XML document that complies with the FormWare specification for XML documents. This requires that you use the <REMOTECFG> property tag to define the JobFlow to which you are submitting the resulting BDF and batch data file. The <REMOTECFG> tag references a CFG file that must exist at the host site in the \FormWare\Remote\Configs folder. The CFG file specifies the JobFlow name and JobFlow task in which the imported XML data should be inserted. Without this information, the resulting BDFs and batch data files may be transferred to the host site, but may not be introduced into the JobFlow for continued processing.

The following example XML document creates a file named "File1" from job "JobA" using the remote configuration file XMLCnfg.cfg:

```
<?xml version="1.0" ?>
<transaction>
    <name>E:\FormWare\files\File1.bdf</name>
    <JobFlowSettings>
        <REMOTECFG>
                <value>XMLCnfg.cfg</value>
        </REMOTECFG>
    </JobFlowSettings>
    <property>
        <name>JOBNAME</name>
        <value>JOBA</value>
    </property>
    <document>
        <name></name>
        < <A> image >
                <name>E:\FormWare\images\JOBA01.tif</name>
                <form>
                        <name>FormA</name>
                        <field>
                                <name>FirstName</name>
                                <value>John                    </value>
                        </field>
                        <field>
                                <name>LastName</name>
                                <value>Doe                  </value>
                        </field>
                </form>
        </image>
    </document>
</transaction>
```

Importing the XML File

The next step is to import the XML file into the FormWare system. There are two import methods. The method you use depends on your organizational requirements.

The first method is to use the Remote Transfer Utility to import XML files in an ad hoc fashion. To use the Remote Transfer Utility:

1. Select the Remote Transfer Utility shortcut from the Windows Start Menu (i.e., **Start > Programs > FormWare Remote**).

2. Select the **Transfer Files** option.

3. Select the **Perform Operation** button**.** A Browse Files dialog displays.

4. Select XML Files (*.xml) from the Files of type.

5. Select one or more XML files to import. At this point, the Remote Transfer Utility processes XML files until you stop it.

The second method for importing XML data into a JobFlow is to use a programming API within the FormWareRemote.dll located in the \FormWare\Programs folder. The function prototype is:

```
extern "C" BOOL FWRMTSubmitXMLFile(LPCTSTR lpszXMLFile)
```

where LPCTSTR lpszXMLFile is– the fully qualified name (UNC or drive mapping) to the XML file to import. The function will return 0 on success, otherwise an error occurred. You can retrieve the error type by using:

```
extern "C" FWRMTError FWRMTGetLastError();
```

The following (defined as typedef enum FWRMTError) errors are possible:

Error Type	Description
FWRMT_NO_RMTCONFIG	RmtXfer.ini does not exist or cannot be accessed
FWRMT_LOGIN_FAILED	FormWare IFS login failed
FWRMT_PUTFILE_FAILED	Transfer of the XML data to the FormWare Server failed
FWRMT_XMLIMPORT_FAILED	XML Import client failed to import the transferred XML file
FWRMT_TMPFILECREATE_FAILED	Failed to create a temporary file on the FormWare Server for the transferred XML data
FWRMT_NO_DIRMAN	Unable to initialize the Directory Manager on the FormWare Server
FWRMT_NO_PROP	Unable to initialize the Property Manager on the FormWare Server
FWRMT_TMPREMOVE_FAILED	Removal of the temporary XML file from the FormWare Server failed

You can also get a text description of the error using:

```
extern "C" BOOL FWRMTGetLastErrorMsg(LPTSTR lptstrErrorMessage, DWORD
dwMaxLength)
```

where LPTSTR lptstrErrorMessage is your buffer to contain the text and DWORD dwMaxLength is the length of your buffer.

The following is an example in C++ using Visual C, version 6/MFC:

```
CString szXMLFile = SomeXMLFile;
if(!szXMLFile.IsEmpty())
{
        BOOL b = FWRMTSubmitXMLFile((LPCTSTR)    SzXMLFile);
        if(!b)
        {
                _TCHAR szErr[255];
                FWRMTGetLastErrorMsg(szErr, 254);
                AfxMessageBox(szErr);
        }
}
```

Defining Distributed Completion

Distributed Completion is the remote processing component that enables remote completion operators to work on the Completion mode tasks (e.g., Reject Repair, Verify, Remove Flags). This is the only Distributed Processing component that extracts work from the server, processes it, and returns the modified data to the JobFlow – all via the Internet.

Distributed Completion enables remote operators, to pull data from a host site JobFlow, process it, and return the results back to the host for continued processing. When an operator requests work, all the necessary files for the specified Completion task download to the remote workstation. The operator processes the data using these files. The final step in Distributed Completion has the software automatically upload the completed and modified data into the JobFlow at the host site.

Each Distributed Completion operator must have a unique user INI file that includes configuration parameters for his remote sessions. Among other parameters, the user INI must contain a user name and password. Additional parameters used for Distributed Completion are the pre-fetch count and the information specifying the application files to download during a Remote Sync process. The pre-fetch count determines the number of transactions copied to the local machine at any one time.

The first time a Distributed Completion operator logs onto the FormWare Server, a Remote Sync is performed automatically to copy all of the necessary application files to the remote client workstation. Subsequent connections to the FormWare Server will not automatically perform the Remote Sync process unless you have forced this process programmatically through the remote user INI file using the RMT_FORCE_REMOTE_SYNC=Y parameter. See the Configuring the File Server for Distributed Processing section in Installing & Configuring for more information. Performing the Remote Sync process should be a regular part of a remote user's activities. On subsequent Remote Syncs, only updated files are copied to the local machine, but you should still be aware that the time required to execute the process is a function of the speed of the connection and the size of the updated files.

As work is checked out of the host site for a Distributed Completion operator, the information at the host site is modified to indicate the specific data items are in process. The user name associated with these in process items is the same as the user name entered at login. A Remote Worklist file, <UserName.rwl>, is created on the FormWare Server in the \FormWare\Remote\Worklist folder. A copy of this file is also created on the remote client workstation.

Setting Distributed Completion INI Parameters

The following INI file parameters are important to successfully install Distributed Completion:

Setting Parameters Within the System.ini File

If you are installing FormWare and setting up a Distributed Completion application for the first time, a number of INI parameters must exist within the System.ini file on the FormWare Internet Server:

- LFN=Y - Long filename support (important for file transfers)
- YEAR=4DIGIT - Use a four-digit year within dates.
- Correct format of the Operator's statistics

If you are upgrading an existing installation of FormWare and Distributed Completion, the installation process will check for these settings an add them if necessary.

Setting Parameters Within the Job INI File

You can use the Transfer Enhanced Images (TEI=Y,N) INI parameter to control access to confidential data, such as Name and SSN that is displayed on images sent to Distributed Completion operators. Enhanced images are sent to remote Completion operators for processing instead of the entire image. Use Designer to create snippets of information for viewing. This parameter also deactivates the Full Screen viewing feature in Distributed Completion.

Note: While TEI parameter can be set in any INI file (group, user, system, etc.), it is typically recommended for application within the job INI file.

Setting Parameters Within the User INI File

Use the RMT_FORCE_LOGIN=Y/N parameter in the User INI file to force the user to enter a valid password each time they launch a Distributed Completion session.

RMT_FORCE_LOGIN=Y - Forces the user to type a password

RMT_FORCE_LOGIN=N - Enables Distributed Completion session to on the remote workstation without requiring the remote user to type in a password. The encoded password stored on the remote machine is used instead.

For more information on setting remote user INI parameters, see the Configuring the File Server for Distributed Processing in Installing and Configuring FormWare for more information.

Configuring a Remote Internet Connection

With the correct Internet connection settings, you can create a very efficient Distributed Completion environment. By following the instructions below, users do not need to establish a dial-up connection prior to launching Distributed Completion. Instead, Distributed Completion detects the absence of an active connection and prompts the user to connect, using a default connection.

In addition, these settings allow users to key data off-line. Once work is complete, Distributed Completion looks for an active connection. If one is not found, the user is prompted to re-establish the connection to return the work to the host site. In other words, users can establish a connection, download a block of work, disconnect, perform the work, and re-connect to return the work.

To configure a remote Internet connection:

1. Open Internet Explorer (IE).

2. From the Tools menu, select **Internet Options** to display the Internet Options dialog.

3. Select the **Connections** tab.

4. If you already have dial-up options, they display in the Dial-up settings area. If you do not have any dial-up options, select **Add** to create a new dial-up connection.

5. After adding a dial-up connection, highlight the name of the connection in the Dial-up settings area and select the **Set Default** button.

6. Select the **Always dial my default connection** option. With the default connection selected, select the **Settings** button. IE displays the settings for your default connection.

7. In the Dial-up settings area of the Settings dialog, select the **Advanced** button. The Advanced Dial-up dialog displays.

8. Deselect the Disconnect if idle for option.

9. Select the Disconnect when connection may no longer be needed option.

10. Select **OK**. Continue to select **OK** on each subsequent dialog.

Monitoring the Remote Worklist File

Remote Worklist files (RWL) are created on the FormWare Server wherever a user has files in process remotely. The Remote Worklist files are stored in the \FormWare\Remote\Worklist folder and on the remote client workstation.

To view a Remote Worklist file:

1. Open FormWare Administration and display the contents of the \Remote\Worklist folder in the File Browser.

2. Double-click a file within the Contents area to display the Remote Worklist dialog. It provides the following information:

   - **Workflow Name** – Name of the JobFlow.
   - **Task Number** – JobFlow task number.
   - **Program Name** – Name of the program that runs on the remote client. Currently this is always Complete.exe or Complete.ocx.
   - **File State** - State of the RWL file, which may have the following values:
     o *WorkFlow Checked Out* – The worklist objects have been checked out of the JobFlow.
     o *Transfer to Client failed* – Failed to transfer the RWL and/or worklist objects to the client.
     o *Client has Data* – The client has the RWL and worklist objects.
     o *Client processing* – The client is processing the worklist objects.
     o *Client processed* – The client has completed processing the worklist objects.
     o *Server has data* – The completed data for all worklist objects has been returned to the server.
     o *WorkFlow check in* – A JobFlow check in of the processed worklist objects is in progress.
   - **Work List** - Contains an entry for each transaction defined in this remote worklist. Each transaction has a sequential transaction number and one of the following values identifying the state of the transaction:
     o *Creating RWL file* – The RWL file is being created.
     o *RWL file created* – The RWL file is complete.
     o *Transmitting to client* – The worklist objects are being transferred to the client.
     o *Client has file* – The worklist object was successfully transferred to the client.
     o *Transmitting to server* – The worklist object is being transferred to the server.
     o *Server has file* – The worklist object was successfully transferred to the server.
     o *WorkFlow checking in* – A JobFlow check in of the worklist object is in progress.
   - **Return Code** - Defines the value that the Complete.exe program returned when the processing of this transaction was complete (initially zero).

- **Command Line** - Contains the command line parameters that will be passed to the Complete.exe program for this transaction.
- **Transaction Path** - The fully qualified path of the transaction. In the server version of the RWL, this is the server location. Once the RWL is copied to the local system and processed, the local system copy contains the local system path to the transaction.

When the work is returned, the in process status is updated, the Remote Worklist files are deleted, and the data items proceed to the next task in the JobFlow. At that time, the remote user can download more work as desired.

Note: You can also view the remote worklist file from the command line. Use the following syntax: FWRMTVu <fully qualified path>\<filename.rwl> (e.g., FWTRMvu C:\fwshare\formware\remote\worklist\jsmith.rwl).

Defining Distributed WorkFlow

The Distributed WorkFlow process enables you to process data items on two FormWare systems that are connected only via the Internet. To do this, you scan and process data items in one system, and then transfer them to another system where additional processing occurs. Hence, Distributed WorkFlow is like inserting one JobFlow into the middle of another.

Distributed WorkFlow uses the following systems:

- **Primary System** - The master FormWare installation where you initially scan in data items.
- **Secondary System** - A remote FormWare system to which you transfer data items and where those data items are subsequently processed.

Distributed WorkFlow also uses configuration settings from the Remote Configuration file (CFG). See the Defining a Remote Configuration File section for instructions on how to set up a CFG file. See the following diagram:

Distributed WorkFlow
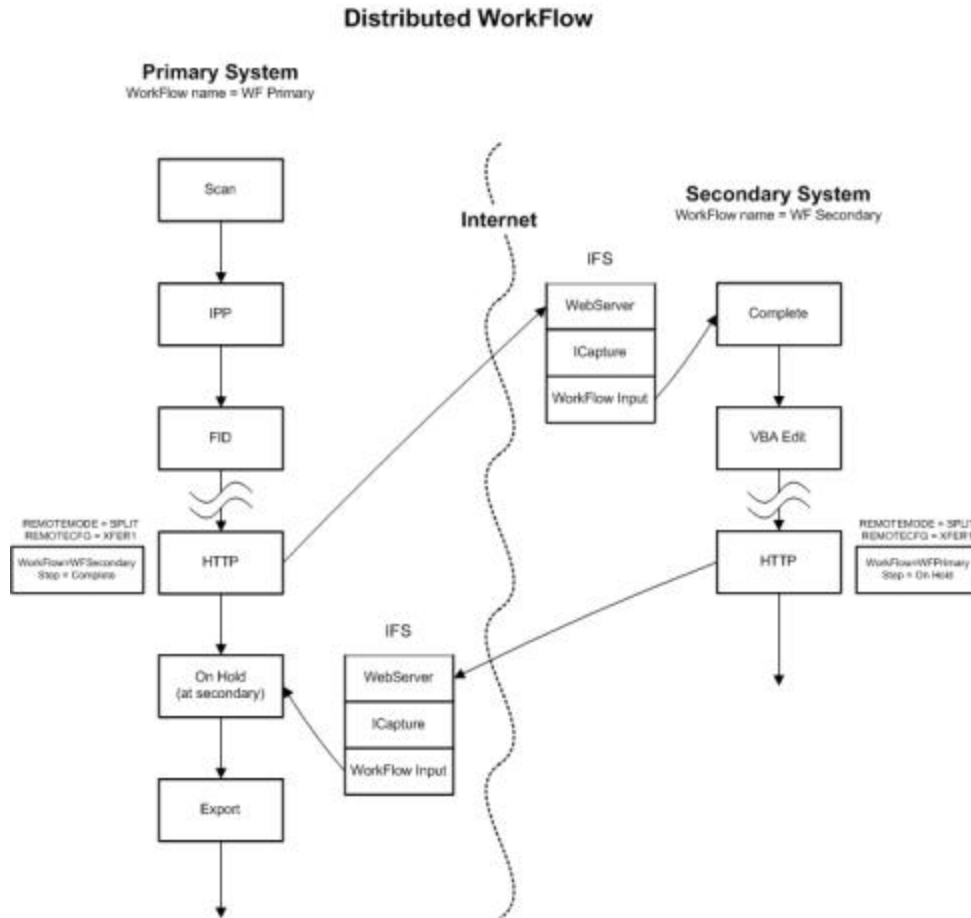
Once batches finish processing on the secondary system, you can transfer them back to the primary system. The primary system picks up the work right where it left off in the original JobFlow.

Distributed WorkFlow is different from Distributed Completion. Distributed Completion keeps batches within a single JobFlow and remote computers connect using the Internet to process work as if the remote machine were just another processing station. Distributed WorkFlow allows you to export work from one JobFlow to another while maintaining enough information across the transfer to seamlessly bring the work back to the original system at a later time.

The main property that differentiates Distributed WorkFlow processing from other FormWare processing is the incorporation of an Internet transfer component that controls sending data from one system to another. This component is called HTTPClnt or HTTP. Each installed system that sends work to another system must use HTTP to perform the transfer. In addition, each system that receives work from HTTP must be running an IFS.

You can use the Split and Join functions for batches across multiple, distributed file system installations. The primary site uses HTTPClnt to transfer batches into the FormWare Internet Server at the secondary site (i.e., "split" operation). The primary site then keeps the batches in a holding queue waiting for them to come back. The secondary site further processes the batches and then uses HTTPClnt to transfer them back to the primary site (i.e., "join" operation). Upon joining the batches, the primary site routes them from the holding queue to the next step in the JobFlow.

Setting Up the Primary WorkFlow System

There are two main steps in configuring Distributed WorkFlow: setting up the primary system, and setting up the secondary system.

Creating an HTTP Task

To set up the primary system, you must create a custom task for the HTTP component in the primary system at the point where work will transfer to the secondary system. To create this task:

1. Open the JobFlow in JobFlow Designer.

2. Select the **Custom** task toolbar button to add a custom task to the JobFlow.

3. Double-click on the task to display the Properties dialog, General tab.

4. In the **Client Program Name** field, type: HTTPClnt

5. In the **Client Parameters** field, type the following:

6.   -REMOTECFG=Transfer1 -REMOTEMODE=SPLIT

7. Where **Transfer1** is the name of the remote configuration file. See the Configuring HTTP Properties section for more information about these parameters.

8. In the **Task Description** field, type "HTTP Transfer to Secondary" or another meaningful name.

9. In the **Icon Name** field, type the name of a valid BMP file.

10. Select the **Save** button to save your task.

Configuring HTTP Properties

You must configure the REMOTECFG and REMOTEMODE properties to run HTTP. You can configure these two properties using the command line arguments, also known as the client parameters, or in the job INI file or even programmatically using VBA.

Note: You must use a text editor to edit the remote configuration files used by the HTTP task. Do not use the Remote Configuration Properties dialog to edit this file because it uses the names of local JobFlows and tasks. Because the target JobFlow and HTTP task are at another location, this program will generate the wrong information.

The REMOTECFG property specifies the name of a remote configuration file that determines where and how data items are inserted into the receiving system. The remote configuration file specifies the JobFlow name, task, and priority where batches are transferred.

Caution: The file specified by REMOTECFG must reside on the system sending the data, where HTTPClnt is running. The contents of this file must refer to a JobFlow on the receiving system, where IFS is running.

The REMOTEMODE property can have one of two values: SPLIT or JOIN. The HTTP component sending work from the primary system to the secondary system uses the SPLIT value. The HTTP component returning work from the secondary system to the primary system uses the JOIN value. You do not need to specify this property if work is not returned to the primary system. Following is an example of a JOIN file.

```
SEND.CFG
JobName=REGIO418
JobflowEnabled=Y
JobflowName=labels
```

```
JobflowTask=2
JobflowPriority=1
ImageBaseDir=C:\FWShare\FormWare\images
RETURN.CFG
JobName=REGIO418
JobflowEnabled=Y
JobflowName=Distributed
JobflowTask=36
JobflowPriority=1
ImageBaseDir=c:\FWShare\FormWare\images
```

Creating a Holding Task

Create a second custom task in the JobFlow immediately following the HTTP task in the primary system. This task acts as a placeholder for data items processing in the secondary system. Names of the data items that transfer successfully are routed to this step and await the return of the data items from the secondary system. No FormWare component processes this holding queue. Instead, when the data items return, information they contain allows the IFS subsystem to locate their names in the holding queue and then immediately forward them on in the system.

If you are not returning work to the primary system from the secondary system, you do not need to create the second custom step.

To create this task:

1. Open the JobFlow in JobFlow Designer.

2. Select the Custom task toolbar button to add a custom task to the JobFlow.

3. Double-click on the task to display the Properties dialog, General tab.

4. Leave the Client Program Name and Client Parameters fields empty.

5. In the Task Description field, type "Batches at Secondary" or another meaningful name.

6. In the Icon Name field, type a valid bitmap (BMP) file name.

7. Select Save to save the task.

Setting Up the Secondary WorkFlow System

Setting up the secondary system is similar to setting up the primary system. To set up the secondary system, you must create a custom task for the HTTP component in the secondary system at the point where work will transfer back to the primary system.

1. Open the JobFlow in JobFlow Designer.

2. Select the **Custom** task toolbar button to add a custom task to the JobFlow.

3. Double-click on the task to display the Properties dialog, General tab.

4. In the **Client Program Name** field, type: HTTPClnt

5. In the **Client Parameters** field, type the following:

    ```
    -REMOTECFG=Transfer2 -REMOTEMODE= JOIN
    ```

6. Where Transfer2 is the name of the remote configuration file.

7. In the **Task Description** field, type "HTTP Transfer to Primary" or another meaningful name.

8. In the **Icon Name** field, type the name of a valid BMP file.

The REMOTECFG property and REMOTEMODE property are also used with HTTP in the secondary system. The use of these properties is nearly identical to that for the primary system, except the file specified by the REMOTECFG property must reside on the secondary system and must contain references to a JobFlow on the primary system.

Note: When you use HTTP to return data items to the primary system (REMOTEMODE=JOIN), the step listed in the remote configuration file must be the holding step of the primary system. If a returning data item is not found in this step, an error will occur.

Defining FormWare Filler

FormWare Filler is a data entry application that enables you to execute a non-imaging application and send the data to a central site for additional processing. FormWare Filler has an easy-to-use interface that enables you to enter new data or add records to an existing batch. Typically, there are no paper documents associated with a FormWare Filler session. If data was first captured by paper, the paper is not required at the central site for further processing.

When data is ready to send to the host site, the operator uses a dial-up connection to transfer data across the Internet. Once received by the host site, the data can be automatically added to a JobFlow at any logical point, such as Export.

For example, a marketing department uses personal interview surveys to gather statistical data for an upcoming campaign. Using a predefined form in FormWare Filler, researchers can promptly input their findings and transmit the data for processing. The timely processing of this information provides fresh data for analysis.

FormWare Filler uses remote configuration files (CFG) to specify the job name and JobFlow for Distributed Processing files (CFG). You create remote configuration files at the host site in the \FormWare\Remote\Configs folder. CFG files are copied to a remote user's local machine during the Remote Sync process based on the RMT_RMTCFG= parameter in the user INI file.

Defining a Remote Configuration File

The remote configuration files (CFG files) contain job, JobFlow, and WorkFlow information necessary for the remote data to be correctly introduced into the JobFlow or WorkFlow on the FormWare Server. The following distributed processes use information within the CFG files: FormWare Filler and Distributed WorkFlow. You define CFG files once on the FormWare Server. Then, when a remote client logs on to the system, these files are copied to the remote client using the Remote Sync utility. See Understanding Remote Sync in the Configuring the File Server for Distributed Processing section of Installing & Configuring FormWare for more information.

To create or edit a CFG file:

1. Open Administration.

2. In the File Browser, select the \Remote\Configs folder.

3. Right-click in the Contents area of the \Remote\Configs folder and select **New** from the context menu.

   Or,

   Double-click on an existing CFG file in the Contents area. The Remote Configuration Properties dialog displays.

4. In the **Job Name** field, select the job associated with this configuration file. When a remote user starts a FormWare Filler session and selects a CFG file, the **Job Name** determines which job (JDF) is presented to the operator.

5. In the **Prefix Strings** area, set indicate the location on the FormWare Server where to put certain files coming from remote workstations. The Internet FormWare Server (IFS) references this path information to direct image, batch data, and BDF files to the proper location on the FormWare Server. The following processes use this information: FormWare Filler (for BDF and batch data files) and Distributed WorkFlow (for BDF, image, and batch data files).

    o   In the **Image** field, type the location that image files coming from remote workstations are put on the FormWare Server. The default path is the \FormWare\Images folder. The path within **Image** field does not apply to FormWare Filler processing.

    o   In the **Batch** field, type the location that BDF and batch data files coming from remote workstations are put on the FormWare Server. The default is the \FormWare\Files folder.

6. In the **File Transfers** area, select from the following options to exclude specific file types from transferring from a remote site back to the FormWare Internet Server. By default, all file types are transmitted from remote workstations back to the FormWare Server, however, because these files already exist on the server site, transferring them back can be a huge waste of time and transfer overhead. By excluding file types that are not required on the server or that may already exist there, you can reduce the file transfer volume from remote workstations and increase file transfer rates and efficiency.

    o   **Exclude Batch Data File** - Excludes BDF files.
    o   **Exclude Image File** - Excludes TIFF, JPEG and other image files.
    o   **Exclude OCR File** - Excludes Optical Character Recognition (OCR) files.
    o   **Exclude Enhanced Image** - Excludes processed image files (SUP files).
    o   **Exclude Thumbnail Image** - Excludes files containing thumbnail-sized images.
    o   **Exclude Color Image** - Excludes JPEG image files.

7. Select the **Enable Workflow** check box in the **WorkFlow Settings** area to specify the JobFlow and JobFlow task to which batches will be added when they are transferred from remote clients to the FormWare Server.

8. In the **Add to WorkFlow** field, select the name of the JobFlow to which the new BDF files will be assigned.

9. In the **Task** field, select the name of the JobFlow task to which the new BDF files will be assigned.

10. In the **Transactions Arguments** field, type any command line arguments you want to pass to the FormWare component during this JobFlow task.

11. In the **Priority** field, select the number that identifies the priority value assigned to the batches processed with this configuration file. JobFlow uses the priority information to determine the order in which BDF files are processed.

12. Select the **Add** button next to the **Registrations** field to create a registration numbers. These numbers are used to group and monitor BDF files during the JobFlow process. The Registration Numbers dialog displays. Highlight a number and select **OK** to assign that registration number to the new BDF files. To remove a registration number, highlight it and select the **Delete** button.

Note: The fields and settings within the Registrations area only apply to JobFlows on the File System – not to WorkFlows on the Database.

13. Select the **OK** button to confirm your changes and save the CFG file. The Save As dialog displays.

14. If you do not want to accept the default name, type a new name for the CFG file in the **File Name** field.

15. Select the **Save** button to save the CFG file and return to Administration.

Defining Distributed Scan

Distributed Scan uses the Scan Designer and Scan Operator modules at a remote site. Scan operators use the Scan Designer module to create or modify scan configuration (SCG) files. They then use the Scan Operator module to scan documents and create batches and image files, which are then transmitted from a remote workstation, via the Internet, to a JobFlow on the FormWare Server for processing.

See the Defining Scan Settings for more information on Scan Designer and Scan Operator.

See Setting JobFlow Properties for Scanning for instructions on how to create SCG files for Distributed Scan.

The Internet Forms Server (IFS) and the Remote Transfer Utility control the Distributed Scanning processes. Using Distributed Scan, you can send image files and BDFs to a server from remote client workstations in any location. The Remote Transfer Utility locates the BDFs and image files and transfers them through the IFS to the FormWare Server for storage and inserts them into a JobFlow.

Note: A scanner is not required to use the FormWare Scan modules.

Defining eEntry

The eEntry application enables you to perform Completion using image-enabled I-Forms. This includes KFI and OCR character correction, using the IFS. On the server side, content is directed into a workflow from which it is routed to remote clients. On the client side, a remote user obtains available work via the Internet Explorer browser, keys data into an ad hoc I-Form, and returns the finished work to the FormWare File Server. Images are cached locally on the client machine in a temporary user folder and removed when a batch is sent back to the server.

Before implementing eEntry, familiarize yourself thoroughly with Creating an I-Form in the FormWare documentation. I-Forms is a simple form-based application with no image controls where users submit the data they key to the server.

Understanding eEntry Requirements

eEntry uses Internet Explorer v.6.0 as the platform for its client application. Following are the software components required to run eEntry.

- A FormWare license that includes a Name of Client DAT file
- An eEntry licensing DAT file

- Internet Explorer v6.0 or later
- A Web page with links to the FWCGI and IIS servers
- A FormWare enterprise application that includes the following components:
    - An I-Forms installation, including job-specific I-Forms
    - FWCGI (IIS)
    - A FormWare File Server
    - An Internet FormWare Server (IFS)
    - I-Capture service running on the FormWare File Server or IFS

Configuring a Server for eEntry

To install eEntry on as a part of your FormWare enterprise installation, perform your FormWare installation as described in the Installing & Configuring section. On the Select Components screen select the FWCGI checkbox to ensure the installation of all necessary eEntry components. The installation process places the TCImage.cab, FWCGI.exe, and FWCGI.dll in the CGI-Bin folder. The components are thus available via HTTP due to their location under the web root.

Perform the following steps to set up your server to run eEntry.

1. Create a FormWare job on the server that contains images.

2. Create an I-Form that corresponds to your processing needs, including the KFI or OCR character correction task. The I-Form you create can be as simple as a single field for keying in a name or a form with enough fields to capture hundreds of pieces of data. The I-Form enables you to extend data capture to web-based forms. Anyone with Internet access and Internet Explorer v6.0 or later can link to the workflow data and view images on the image control page. See the Creating an I-Form section for more information.

3. Create a login screen for remote users to access eEntry. See the Building a Login Web Page section below for more information.

4. Enable the login for the remote workstation using one of the following two methods.
    - Email the remote operator a URL link for the login screen you created.
    - Embed the HTML link to the login screen on a secure web page and authorize the remote operator to access that web page and link up with the assigned task queue.

Note: In most cases, a simple HTML link button on an existing company web page suffices to enable remote users to login. However, you can also create a discrete web page for the remote link.

5. On the FormWare File Server create a unique user INI file for each user to enable them to log in to the image queue. The INI file sets up assignment or task authority for each remote user. The image queue (mode) must correspond to the task number assigned to a given user. You can assign more than one queue to a remote operator.

    The ALLOWJOBFLOW setting in the user INI file designates the workflow and step or task number assigned to the named user (RMT_USERNAME=<name>). The ALLOWJOBFLOW setting must reflect the task number corresponding to the image queue assigned to the user. The workflow name in the user INI file and the name entered at the eEntry login screen are case sensitive and must be identical to the workflow name identified with the image queue on the server. If the ALLOWJOBFLOW designation fails to correspond exactly to the task number of the assigned queue, eEntry displays the following message when the user tries to

> begin a session: "The queue you are checking working out from is not supported. Check with your administrator" or "Unsupported Function; Must be Completion."

6. Below is an example of an eEntry user INI file:

> Refer to the Understanding Distributed Processing for more information on user INI files.

7. When an eEntry KFI session starts, the HTML generated for the startup web page resizes the browser to full screen. If you want to suppress this resizing, set the INI property TCKI_CLIENT_NORESIZE=Y in an INI file. You can set this INI parameter at the system, job, or user level.

Implementing Field Edits

When you create an I-Form for your eEntry application, you may want to add field edits. eEntry supports the same client-side field edits as I-Forms including Must enter, Must complete, and Field Type (). Refer to Creating an I-Form for more information.

eEntry does not provide any character-level edits in the context of a character event. For example, eEntry does not support client-side edits that require keystroke handling, such as changing lower-case letters to upper case as they are keyed. Moreover, eEntry provides no key beyond image, so the ability to key records without images after the last image in the batch is not available. If you want to execute field edits, you must use JavaScript or VBScript. For more information about field edits refer to the Defining Selective Edits, Setting Field Edit properties section.

Note: Only custom edits need coding. Any custom-edit code you implement must take into account that FormWare edits are written in JavaScript.

Building a Login Web Page

You can set up the eEntry login as a single-step or a two-step process. If you create a single-step login, the login screen must contain two fields for authentication, the user name and password, and two fields for task assignment, workflow name and image queue name or number. If you create a two-step login, the first screen must the two authentication fields, and the second screen must contain the two task assignment fields. The procedure for creating an eEntry login web page below describes the creation of a two-step login.

The instructions for creating a login web page assume these tasks will be performed by a web author and system administrator that are familiar with web servers and HTML. The web author

builds the web page and sends it to the administrator, who would install it on the server. In some cases, the same person may perform both functions.

Perform the following steps to create a login web page for your eEntry application.

1. Using a text editor or an HTML editing tool, create a static web page that contains a login form for sending a user name and a password to the workflow web server.
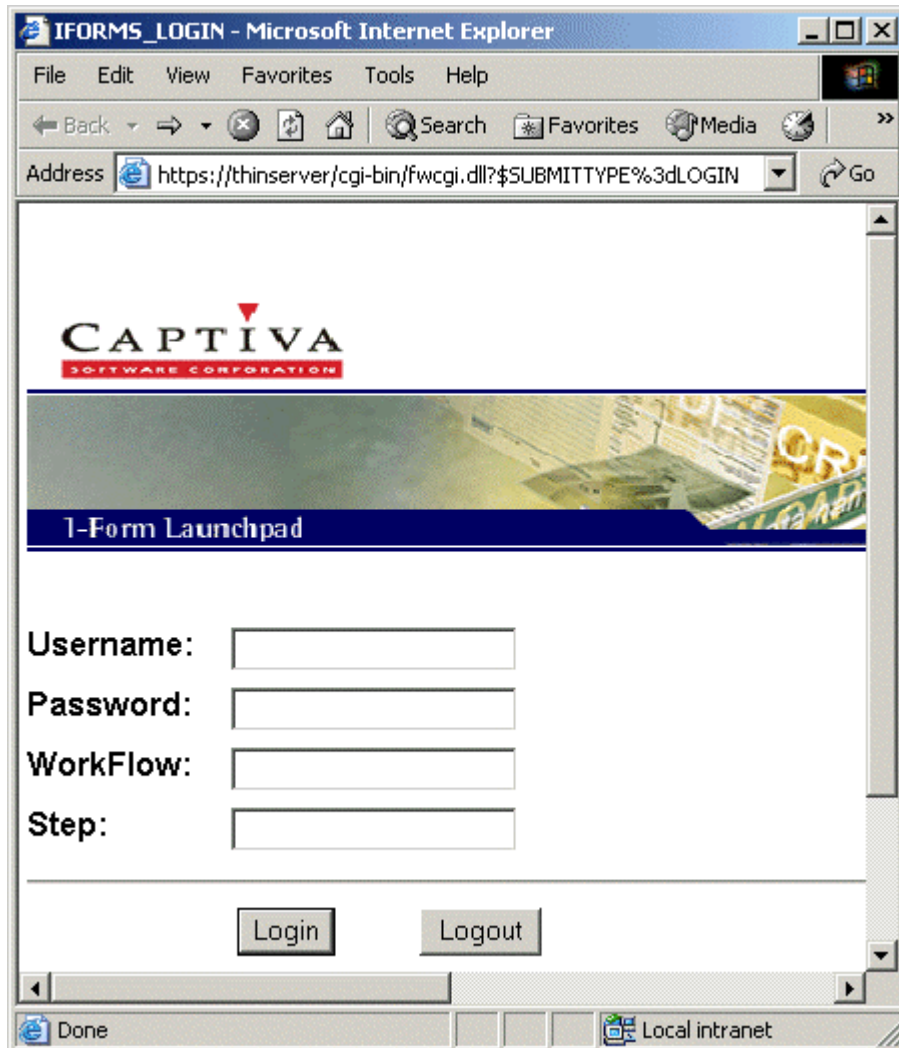
2. Create a second web page that contains a form for sending the workflow name and image queue name to the web server.
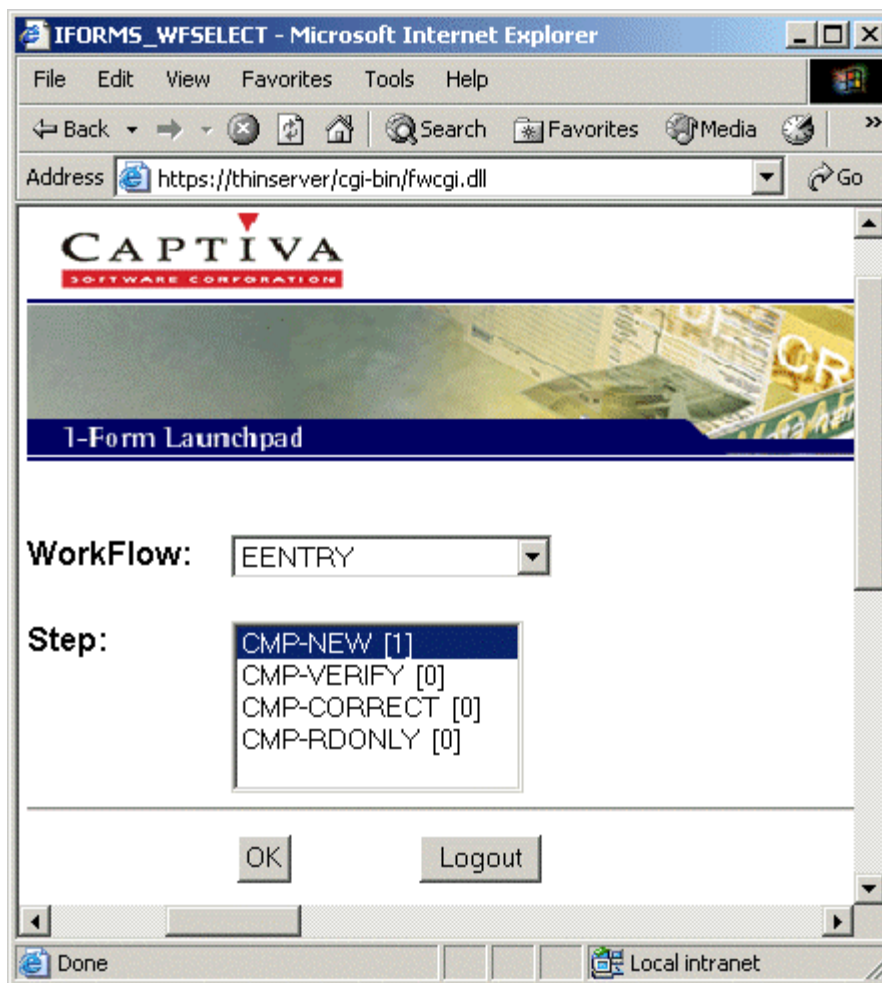
3. Link the login web page to the web page for selecting work. Refer to the sample form definition files (FDFs), I_Forms_Login.fdf and I-Forms_WFSelect.fdf, furnished by Captiva to facilitate this remote access.

4. You can preview the HTML source code behind these FDFs on the Form Properties, I-Form tab in Designer. Designer launches both the browser and the generated HTML so you can see how the form displays in the browser. Additionally, you can view the HTML code in Designer by right clicking on the form you wish to view and selecting HTML from the Export menu.

5. Copy or publish the finished web page(s) to your web server.

6. The web-page installation steps depend on your choice of web server and the architecture of the web server. Many possibilities exist, but generally, you install the web page by copying the HTML file into the directory of web pages on the web server. All parties involved must agree on an installation location to establish the final URL of the page.

Note: Due to the nature of SSL, the URL you use to load the login/workflow selection pages must be https if the job posting URL is https.

7. Save the login web page(s) destination in browser Favorites.

Configuring a Workstation for eEntry

You perform client side installation from a workstation's browser. When you access the I-Forms server, FormWare downloads and installs the necessary components.

Installing eEntry on a Workstation

Remote users must perform the following steps on their remote workstation to set up eEntry.

1. Configure Internet Explorer (IE) v6.0 or later as the browser for the workstation.

2. Browse to the web page that contains the link to the eEntry installer. The person that set up eEntry on the server can provide you with the web page address.

3. Select a download prompt on the web page to launch the eEntry Installer, which auto-installs the following files on the remote workstation:

   - FWIMGLIB.DLL
   - TCKISRVR.DLL
   - VIC32.DLL
   - TCIMAGE.DLL (image-controls file—the only file that gets registered)

4. Use the user name and password provided by the system administrator to log in.

Enabling Workstations to Access the Server

System Administrators must perform the following steps to enable remote users to connect too the server.

1. Set up security.

2. Establish user names and passwords.

Note: You are responsible for configuring and administering security options. SSL is not part of FormWare, but is a function of the web server.

3. Send the applicable data from the <user>.ini file to each user to enable them to log into eEntry. This information to send includes the following items:

   - User name (must be unique only one user at a time can login)
   - Password (to authenticate login to FormWare)
   - JobFlow or WorkFlow name
   - JobFlow or WorkFlow task or image queue

Configuring the Client Environment for the Proxy Server

When you have a remote network of several machines where only one machine has Internet access, you must run a proxy server on the machine with the Internet connection to enable the remaining machines Internet access. The proxy server filters information and passes it between the remote workstations and the Web server. All that is required to support a proxy server on the eEntry client machine are two environment variables—FWProxy_<name> and FWProxy_<port>. Set these two variables only when you are using a proxy server.

Perform the following steps on each client machine you link to the proxy server to configure the proxy name and port.

1. If you are running Windows 2000, from the Start menu select **Settings > Control Panel > System > Advanced > Environment Variables** to display the Environment Variables

dialog. The path to the Environment Variables dialog may differ if you are running a different operating system.

2. In the System variables list area, select **New** to display the New System Variable dialog.

3. In the **Variable Name** field type Name, and in the **Variable Valu**e field type FWProxy_<name>.

4. Select **OK** to confirm your changes and return to the Environment Variables dialog. The new variable displays in the System variables list area.

5. Select **New** to display the New System Variable dialog again.

6. In the **Variable Name** field type: Port Number

7. In the **Variable Value** field type: FWProxy_<port>.

8. Select **OK** to confirm your changes and return to the Environment Variables dialog. The new variable displays in the System variables list area.

9. Use Internet Explorer 6.0 or later to display the eEntry login page and log into the system.

4 Testing

Testing Overview

It is always important to test your FormWare system in a test environment before putting it into production. When you design a forms processing application using images and recognition technology, fine-tuning the parameters for such an application requires testing and re-testing. The Designer module provides an environment where you can design and test the functionality of the forms processing application you create. The FormWare software enables you to test the properties you set in Designer at the job, form, and field levels. You can also test one process at a time or several processes together on each of these levels.

Testing a single process is referred to as discrete testing, while testing several processes simultaneously is referred to as comprehensive testing. Discrete testing is applied to a single image, while comprehensive testing can be applied to a batch of images. Comprehensive testing is conducted on all levels within the Designer module: job, form, and field. Discrete testing is conducted only on the form and field levels.

For example, if you perform a comprehensive test of IPP, Enhancement, and Recognition, the recognition results will differ from the discrete recognition test performed on the Form Properties, Recognition tab. This is because the IPP and Enhancement processes are involved in the comprehensive test.

To perform discrete testing, access the tab on the Properties dialog for the specific process you want to test. For example, if you want to test only Recognition on the form level, then access the Form Properties, Recognition tab and select the Test button.

To perform comprehensive testing, access the Test tab on the Job, Form, or Field Property dialog. You can select multiple processes to execute in the test and specify a batch of images for the test.

See the following sections for more information:

- Testing a Job
- Testing a Form
- Testing a Field

Developing an Acceptance Test Plan

While designing your application, you should also create an Acceptance Test Plan (ATP) to ensure that everyone understands the acceptance criteria for the project. To write the ATP, review the Detail Specification and determine the business rules and systems that will require testing. The benchmark for acceptance is determined by the requirements listed in the Detail Specification. Thus, if the system performs as per the Detail Specification, then the system must be accepted.

The following guideline offers ideas for organizing a successful Acceptance Test.

Acceptance Test Plan Guidelines

Devise an issue list based upon the Detail Specification for tracking problems, defects, questionable results, and feedback coming from those participating in the test. An issue list serves as the test documentation and provides the structure for organizing, scheduling, and managing the acceptance testing.

Create test cases for each section of the application as defined in the Detail Specification. Every JobFlow task requires a separate section for testing. Your team should agree on the test conditions. Consider the following:

- Create an issue list using the Detail Specification as the basis for your tests.
- Create a test section for each JobFlow task.
- Note that redundant tests waste time and resources.
- If something in the specification document is unclear, make sure that it is clarified.

Gathering Acceptance Test Images

As you create the ATP, you need to select forms that match the test criterion. Although production images are preferred, it may be necessary to create documents to meet specific test cases. The advantage of production forms is that they test the software using real situations. However, there may be confidentiality issues or other reasons why actual documents cannot be used. Forms that are purposefully filled out are beneficial because one form can test multiple scenarios. This reduces the number of test images and test batches.

If forms are filled out:

- If the forms are handwritten, multiple people should fill out the forms.
- Use evenly spaced block letters.
- Poorly written forms slow down the testing process and skew testing results due to poor recognition rates.
- If forms are filled out electronically, match the font and point size that is common to production forms. Typed data, cut and placed strategically on the form, should be carefully glued to the form. Tape causes additional lines on the image after scanning. Place the data carefully in the form area that is zoned for imaging.

All forms:

- Scan the test images using the production scanner.
- Cross-reference the form to the test case by placing a number on the form that is visible to the operator in Completion but that will not interfere with Form ID or recognition results. By placing a reference number in an innocuous location on the test form, the tester can easily reference the ATP test case list with the image.

Creating Acceptance Test Batches

After the ATP is written and the test forms are scanned, create test Batch Definition files (BDF). Organize your images in a sequential manner that will allow operators to move smoothly through the ATP during testing. This allows the operator to encounter each test case image as it appears in the ATP. Careful organization of images and test cases within the ATP assists the project team in tracking issues that may arise. The acceptance test is NOT a volume test. Keep batches small (10-20 images) for easier acceptance test management.

Performing Acceptance Testing

Acceptance testing should be conducted in your production environment. Locate the scanner, workstations, JobFlow clients, and printers in one room or in close proximity to each other. This saves time and effort for the participants because feedback can be quickly relayed and acted upon if necessary.

If a FormWare application is currently in production, develop a plan for testing the new application without impacting the production system. This may involve an isolated test environment requiring additional licenses, or defining a separate testing environment on the FormWare File Server. In either case, this situation requires careful and thorough planning prior to application installation to avoid licensing, networking, and other issues that could occur.

Also, plan a volume stress test prior to using the system in production. It is not possible to stress a system during development and outside of the production environment.

- Reserve an area and workstations for testing
- Use production hardware
- Duplicate the production environment as closely as possible.

Note: If other software applications (such as Microsoft Word or Excel) are intended to be installed on the production hardware and available during production processing, they must be installed prior to the acceptance test to validate peripheral software compatibility.

- Perform a volume stress test on all of the document types used in production. Volume testing can detect unexpected hardware, software or volume related problems that were not apparent in the development environment.

Everyone participating in the acceptance test should have a clear understanding of their role in the testing process and knowledge of the FormWare system and processes. During testing, you should use as many FormWare trained operators as possible to ensure the acceptance test is completed in an accurate and timely manner. The following people should be involved in the ATP:

- Operator and ATP administrative staff should:
  - o Be familiar with the current system and production environment.
  - o Understand data field specifications for testing field edit validity.

- o   Have the testing and analytical skills necessary to determine if the software is performing per the ATP.
- Export testers who can verify the export file and who are familiar with the export layout and requirements.
- System Administrative staff who are familiar with current operations requirements.

Testing a Job

Use the Test tab on the Job Properties dialog to run a comprehensive test of the properties set for the job on an image or a group of images. You can select a number of processes to run the test on and can create a JobFlow of the selected processes.

To test your job:

1. Open the job in Designer.
2. Display the Job Properties dialog.
3. Select the Test tab.
4. Select the check boxes for the processes you want to test.
5. Select the **Scan** check box to scan an image or group of images or select the **Browse** button in the BDF area and locate a BDF to use for testing.

Caution: Running the comprehensive test against a BDF can alter the BDF and the images in that BDF. We recommend that you copy all image files in the BDF to a SAVE folder before executing the test.

6. Select the **Test** button to run the process with the current job. Each module loads and executes against the BDF. If you select the Completion testing option, the Completion module displays the results of the other processes. To return to the Designer module, you must exit from the Completion module.
7. If you encounter problems with the job, change the job properties and retest. You can continue to test and retest the job properties until all functions are correct.

Testing a Form

Use the Test tab on the Form Properties dialog to run a test of the parameters set for the data form on a group of images. You can select several processes on which to run this comprehensive test at once. To test an individual process for the data form on the setup image, select the tab for that process.

To test the form properties:

1. Open the job and form in Designer.
2. Display the Form Properties, Test tab.
3. Select the check boxes for the processes you want to test.
4. In the BDF area, select the **Browse** button to select a BDF for the test.
5. Select the **Edit** button if you need to modify the BDF in a text editor.

Caution: Running the comprehensive test against a BDF can alter the BDF and the images in that file. We recommend that you copy all image files in the BDF file to a SAVE folder before running a comprehensive test.

6. Select the **Test** button to run the comprehensive test and create a set of test images from the images in the BDF. These are temporary images used only for the form-level test. If you select Recognition, results are then displayed in the Completion module.

7. If you encounter problems with the data form, change the form properties and retest. You can continue to test and retest the form properties until all functions are correct.

Testing a Field

Use the Test tab on the Field Properties dialog to test the parameters for the current data field on an image or a group of images. You can select several processes to test at once. You can also test an individual process for the field on the setup image, from the tab for that process.

To test the field properties:

1. Open the job and form in Designer.

2. Select the field you want to test.

3. Display the Field Properties, Test tab.

4. Select the check boxes for the processes you want to test.

5. In the BDF area, select the **Browse** button to select a BDF for the test. The BDF selected must have the name of the current job in the JOBNAME= parameter.

6. Select the **Edit** button if you need to modify the BDF in a text editor.

Caution: Running the comprehensive test against a BDF can alter the BDF and the images in that file. We recommend that you copy all image files in the BDF file to a SAVE folder before running a comprehensive test.

7. Select the **Test** button. Progress dialogs for each of the selected processes display. A set of test images is created from the images specified in the BDF. These images are created temporarily and only for use with the field level test. If you select Recognition, results are then displayed in the Completion module.

8. If you encounter problems with the field test, change the field properties and retest. You can continue to test and retest the form properties until all functions are correct.

5 Using

Using Overview

Once your FormWare application has been designed, tested, and placed in a production environment, operators and administrators will follow the steps in this section to use your application. The information in this section includes:

- Using the Scan Operator
- Setting Up JobFlow Workstations
- Adding Batches to a JobFlow
- Running Manual IQC
- Performing Manual Form ID
- Running Recognition
- Performing Completion
- Running a Document Assembly Application
- Using Distributed Processing
- Performing eEntry

You can also use the steps in this section to test your application before putting it into production to ensure that it is working the way you intended.

Using the Scan Operator

The following sections contain basic information on using the Scan Operator module. Using the Scan Operator you view JPEG images as well as black and white TIFF images during the file import process.

Starting the Scan Operator

To start the Scan Operator:

1. Open FormWare Administration.

2. Select the **Scanner** button on the Administration toolbar. The Scan Operator dialog displays.

3. Select the **Config** button to select the scan configuration file (SCG) you will use in your scanning process. The Open dialog displays.

4. In the Open dialog, select the SCG file for this process and select the **Open** button. See the Setting Scanner Properties section for instructions on creating an SCG file.

5. If you selected the **Edit on Start** option for the Job Name, Comment, or Priority fields on the JobFlow dialog in Scan Designer, the Job Properties dialog displays.

6. Edit the **Job Name**, **Comment**, and **JobFlow Priority** for the current batch. Only the properties for which you have selected the **Edit on Start** option will be active.

7. Select **OK**. If you have selected the **Edit on Start** option for any of your string segments on the String Segment Properties dialog, the Edit on Start dialog for the string segments displays.

8. Edit the starting **Value** of each string segment as appropriate. None of the other fields on this dialog are active. Select **OK** to save your settings and return to the Scan Operator dialog.

9. Select the **Start** button to start the scanning process with your settings. If you need to stop scanning at any time, select the **Stop** button. The scanned images display in the Image Log area at the bottom of the Scan Operator dialog. The Image Log contains the following columns of information:

    - **Image** – Individual image paths and file names.
    - **Imprint** – Characters imprinted on the document or image.
    - **Batch** – Individual batch paths and batch names.

10. Select the **Exit** button to close the Scan Operator dialog and return to Administration. The Image Log is emptied when you exit Scan Operator.

Viewing a Scanned Image

Once you have scanned or imported images, you can view them in the Image Viewer area. To view an image:

1. Scan or import files using the Scan Operator.

2. In the Image Log area at the bottom of the Scan Operator dialog, click on an image file to load that image in the Image Viewer area.

3. To view different portions of the image, right-click on the image and drag with your mouse to the area you want to view.

4. To resize the Image Viewer area, place your cursor at the edge or corner of the Scan Operator window until the double-sided arrow displays. Select the arrow and drag the edge of the window until it is the size you want. You can also stretch or shrink the vertical size of the Image Viewer area by using the leftmost blue arrow buttons on the toolbar, **Stretch Image** and **Shrink Image**.

5. Use the buttons on the Image Viewer toolbar to change the view of the image without changing the image file. You can also print the image. Buttons on the toolbar (from left to right) include:

    - **Previous Page** – Display the previous image in the series.
    - **Next Page** – Display the next image in the series.
    - **Thumbnail** – Open a thumbnail-sized window that contains the entire image. The black area of the thumbnail corresponds to the area of the image displayed in the viewer. Use the mouse to move the black area to change the portion of the image viewed in the Image Viewer. Select the **Thumbnail** button again to close the Thumbnail window.

Note: If the image is not visible (thumbnail is gray), your color palette settings are incorrect. The thumbnail can only function with a setting of 256 colors – not 16 colors, high color, or true color settings.

- **Zoom In** – Enlarge the image in the Image Viewer.
- **Zoom Out** – Shrink the image in the Image Viewer. You can also left double click in the Image Viewer area to zoom out fully. This produces the same effect as using the **Fit to Window** button.
- **Fit to Window** – Resize the image to fit the size of the Image Viewer window.
- **Zoom** – Percentage size at which your image displays. You can also do this by left clicking and dragging in the Image Viewer area to define a zoom region. Define a small region to create a high zoom effect. You can pan a zoomed image by right clicking and dragging in the direction of the desired pan.
- **Rotate Left** – Rotate the image 90 degrees to the left. You can also do this by pressing [Shift]-[Ctrl] and left clicking in the Image Viewer area.
- **Rotate Right** – Rotate the image 90 degrees to the right. You can also do this by pressing [Shift]-[Ctrl] and right clicking in the Image Viewer area.
- **Flip** – Rotate the image 180 degrees.
- **Stretch Image** – Vertically increase the size of the Image Viewer area.
- **Shrink Image** – Vertically decrease the size of the Image Viewer area.

6. Select **Properties** from the View menu to display the name, height, and width of the image in the Image Viewer.

7. Select **Exit** from the File menu to close the Image Viewer.

Changing the Scanner Source (Kofax)

Using the Scan Operator module you can select a different scanner source device than the one selected for the SCG file in Scan Designer. To choose a new scan source:

1. Open Scan Operator.

2. Select the **Config** button and select an SCG file that uses a scanner as its source for image files.

3. Select the **Source** button to display the Select Scan Source dialog.

Note: This button becomes active when you open an SCG file. If you interrupt a scanning session using the **Stop** button, the **Source** button changes to inactive and the Scan Operator automatically uses the previously chosen source when you resume the scanning session.

---

4. In the **Source** field, select a new scanner source from the list of available scanners. The list is compiled from Scanner Source Settings in the Control Panel, KSM applet.

Note: You can now select "None (No Scanner)" as an option if you only want to import images. Use this option when you do not need a scanner to import images.

---

5. Select the **Properties** button to view the properties of the selected scanner. The Select Scanner dialog displays. The fields on this dialog are read-only. Select **OK** to return to the Select Scan Source dialog.

---

6. Select the **Advanced** button on the Select Scan Source dialog to set advanced features for a source device, including image enhancement, image formatting, paper size standards, and extensions. Your scanner must be equipped with these features for you to use them. The Advanced Source Properties dialog displays.

7. In the Image Enhancement area, select the **Enable** check box to provide access to the image enhancement algorithms available on Adrenaline hardware accelerators. This option is available to you only if you are using Adrenaline hardware accelerators. Additionally, image enhancement is not available when using scanners that return compressed images, such as Kodak scanners.

8. If you enabled Image Enhancement, select an algorithm from the list. The image enhancement algorithms are only available for source devices based on Adrenaline hardware with run time v3.10 and above. Available algorithms are:

    *   Descreening – A shade removal algorithm that is useful for removing halftones from photographs.

    *   Image Smoothing – An image-smoothing algorithm that is useful for smoothing characters and jagged lines on your image.

9. In the Image Formatting area, select the **Enable Width Alignment** check box to force the width of an image to be aligned on a specific boundary. This is useful when images are rotated 90 degrees, and the application requires image width to be even multiples of 8 or 16. Options are:

    *   Pad Width to Byte Boundary – Pads the image width up to the next multiple of eight.

    *   Pad Width to Word Boundary – Pads the image width up to the next multiple of 16.

10. In the **JPEG Compression Quality** field, select the image quality to be retained when images are compressed to JPEG format. This is useful if the sizes of your JPEG files are large and you can trade image quality for smaller files. You can specify any value from 0 to 100. Smaller values result in smaller files but reduce image quality. Larger values result in better image quality but larger compressed file sizes. The default setting is 75.

11. In the Paper Sizes area, select a paper size standard for the paper you use when processing. The paper size options are only available for source devices based on Adrenaline hardware with runtime v3.10 and above. Options are: **Kofax Default** or **ISO Standard**. The default is Kofax Default.

12. Select **OK** to save your settings and return to the Select Scan Source dialog.

Note: The Extensions area is not currently used.

Changing the Scanner Source (ISIS)

Using the Scan Operator module you can select a different scanner source device than the one selected for the SCG file in Scan Designer. To choose a new scan source:

1. Run Scan Operator.

2. Select the **Config** button and select an SCG file that uses a scanner as its source for image files.

3. Select **Open**.

4. Select the **Source** button to display the Scanner Selection dialog.

Note: The **Source** button becomes active when you open an SCG file that designates a scanner as the file source. If your SCG file designates a file or fax source for your image files, the **Source** button remains inactive and the Scan Operator automatically uses that source during the scanning process.

5. In the Scanner Selection window, select a new scanner from the list of available scanners.

Note: You can now select "None (No Scanner)" as an option if you only want to import images. Use this option when you do not need a scanner to import images.

6. Select **OK** to accept the highlighted scanner or **Cancel** to exit this dialog.

7. Select the **Properties** button to view the properties of the selected scanner (Scanner Make, Scanner Model, Primary and Secondary Accelerators, and other accelerator information) The properties fields are read-only. **OK** is unavailable, but you can select **Cancel** to return to the Scanner Selection dialog.

Note: If your selected scanner has no user-definable installation or configuration options, a message displays advising you to refer to your scanner manual for configuration options.

Scanning with a Modified SCG File

If you modify an SCG file after using it to scan, the Scan Operator displays the following warning dialog the next time you open that SCG file. This warning dialog asks if you want to use the scan state file FormWare created during a previous scanning session before you modified the SCG file.

- Select **Yes** to sequentially continue with the batch and image naming in progress.
- Select **No** to reset the state file to the initial values in your SCG file; FormWare resets all of the string segments to their minimum values.

If you do not wish to resume with the string segments set back to their minimum values, set your string segments to **Edit on Start** in the SCG file.

Resuming a Scanning Session

Use Scan Operator to resume a scanning session after an interruption. If you stop a scanning session in the middle, it will continue with the correct image, batch, and imprint string when you restart it. You do this by using a single state file in the FormWare directory. This state file saves information regarding the state of each machine currently running Scan Operator.

Note: Resuming an interrupted session applies only to scanning sessions. It is not available for importing sessions.

To stop and restart a scanning session:

1. Open Scan Operator.

2. Select the **Config** button and select an SCG file.

3. Select the **Start** button to begin your scanning session.

4. After your scanning session has begun, select the **Stop** button to interrupt the session and exit the Scan Operator.

5. Reopen Scan Operator and the SCG file you selected in step 2.

6. If you set any variable to **Edit on Start** in the SCG file, a dialog displays and prompts you to enter a new value.

7. Select the **Start** button to resume your scanning session. The scanning session continues using the correct image, batch, and imprint string.

If you deleted any images before you interrupted the scanning session, the deleted names are not reassigned to other images when you restart the scanning session. The ability to resume a scanning session also lets you interrupt one scanning session, run a second scanning session, and subsequently restart the first scanning session with the correct image, batch and imprint string.

Recovering from a Paper Jam During Scanning

If you experience a paper jam while scanning a set of documents, the Scan Operator module allows you to delete any bad images created. If you are using a duplex scanner, you can delete two images.

Scanning Color Images

Using Scan Operator, you can scan color images. Scan Operator inserts an "FW_IMT=COLOR" tag in the BDF in front of the names of color images. If page, document, or batch tags exist in the BDF and all images within a given batch are color images, then only a single tag is written at the beginning of each batch to identify all images within that batch as color images. The tag is not written to the BDF if the color image is imported rather than scanned in from a paper document.

Setting Up JobFlow Workstations

Understanding the JobFlow Client

The JobFlow Client controls the workflow of your forms processing application. The following discussion describes how a typical JobFlow works in a production environment. After you design your data forms and JobFlow, the job is ready for production. The JobFlow Client moves work through each task in the JobFlow.

During the scan process, operators scan groups of paper forms to create TIFF image files. Typically, the operators group the forms into batches of 50 to 100 paper forms and then scan them. The scan process creates a BDF, which is an ASCII file that contains the name of the job and list of the image files.

The JobFlow Client sends the BDF and all related images through Image Pre-Processing (IPP), Form ID, and Image Enhancement processes to perform such tasks as deskewing, stretching, registration, and removals on each image in the batch. This produces cleaner images and improves readability by the recognition engine. The JobFlow Client calls the first recognition engine to perform recognition, which can also run engine-specific rules

The FormWare recognition process receives a registered, removed page image and its related rule specifications, recognizes image data using all requested engines, arbitrates the results between the engines, and returns the results to the JobFlow Client. Recognition produces an OCR file, which contains the image data captured by the recognition engine. If you choose to use voting, or if your job requires multiple recognition engines, the JobFlow Client calls the next recognition engine to perform Recognition and modify the OCR file.

The JobFlow Client sends the OCR file to the Completion module for Reject Repair, Remove Flags, and Verification (or other modes of Completion you defined). Operators repair and check low confidence characters and recognition errors. You can also create edit programs to further validate the data. Finally the data is exported and the final JobFlow step, Clean Up, removes the files that are no longer necessary.

Adding Batches to a JobFlow

Once you create a JobFlow system, each workstation must be logged into the JobFlow system to be assigned work. Before a workstation operator can access work, transactions must be assigned to the JobFlow tasks within the system and BDFs must be created.

Creating Batch Definition Files

FormWare requires that a Batch Definition file (BDF) be created when image processing is performed. BDFs can exist anywhere in the FormWare root directory, although the \FormWare\Files directory is the default location. A BDF is a text file with the following structure:

```
JOBNAME=HCFA1500
F:\FORMWARE\IMAGES\TEST0001.TIF
F:\FORMWARE\IMAGES\TEST0002.TIF
F:\FORMWARE\IMAGES\TEST0003.TIF
```

Note: You can replace the full path with the parameter %FORMWARE%. When you use this parameter, the software uses the path specified by the FormWare environment variable. This allows you to move images from server to server without having to change the directory information in the BDF file.

There are a couple of different methods for adding batches to JobFlow tasks. The following sections describe each of these methods.

Adding Parameters to a BDF

You can add the following parameters to a BDF:

- IMAGES – Specifies the path for images in the BDF rather than listing the complete path for each image.

```
JOBNAME=HCFA1500
IMAGES=F:\HCFA\243
OCRDATA=J:\HCFA\243
TEST0001.TIF
TEST0002.TIF
TEST0003.TIF
```

- OCRDATA – Specifies the location of the Recognition results files if they are not in the Images folder.

- Form number – Number that identifies the image with a specific data form in the job. Usually, this number has a value of 00 to 99. However, if the FEX=Y parameter is set, then the number can be from 00001 to 16000. Form numbers are typically written to the BDF during Form ID, but this can also occur during the Identify Images mode of Completion or Image QC. The form number is offset from the image name with a colon (:) as shown below:

```
JOBNAME=HCFA1500
01:%FORMWARE%\TEST0001.TIF
02:%FORMWARE%\TEST0002.TIF
04:%FORMWARE%\TEST0003.TIF
```

- Additional data – You can place additional data in the BDF using the Scan Operator (e.g., Document Control Number) or a program. You can insert up to 1,024 characters of data after the image name. If the Read Page Data option is selected for the job, then this data is pulled into the data form during Completion. See this example of a BDF with additional data:

```
JOBNAME=HCFA1500
F:\FORMWARE\IMAGES\TEST0001.TIF 12351515JOHN T. THOMAS
F:\FORMWARE\IMAGES\TEST0002.TIF 23423434JOANNE S. ANDERSON
F:\FORMWARE\IMAGES\TEST0003.TIF 23423190SUSAN SMITH
```

Creating Batches Using the Scan Module

The recommended method of adding batches to the JobFlow system is to use the FormWare Scan Operator module to create images, group them into batches (BDFs), and add those BDFs to a specific JobFlow task during the scanning process.

When you generate images using the scan processes, whether by scanning documents or importing images, they can be grouped in BDFs and added to a JobFlow. See the Starting the Scan Operator section for steps on scanning images to create BDFs.

Creating Batches Using the JFBatch Program

Another way to add batches to JobFlow is to use the JFBatch.exe program. This program is designed for use with custom scanning software or by integrators who develop and market custom scanning software.

When you create a BDF, you can add it to a JobFlow queue by calling the JFBatch program. Use the following syntax:

```
JFBATCH /f=<JobFlow Name> /n=<task #> /p=<priority #> /b=<path & name of
BDF> /R=<#,#,#,#,#,#,#,#> /a="<Batch Arguments>"
```

The /f, /n, and /b switches are required. The specific order of the switches is not important, but case sensitivity is important. The /R= switch is used for registration numbers. You can assign up to eight registration numbers for each batch.

Use the parameter /INTERACTIVE parameter with the JFBatch program to display a message if a batch fails to be added to a JobFlow queue. See Parameters in the Supplementary Information section for more information on the /INTERACTIVE parameter.

If you choose to display a message, be aware that this can interrupt production. If you are calling the JFBatch program from another program, FormWare returns an exit code. The exit codes are listed below:

Num	Error Codes
0	JFBATCH was terminated by the operating system
1	JFBATCH successfully added the batch to the JobFlow queue.
2	JFBATCH was terminated due to JobFlow time-out error
5	JFBATCH Initialization Error Error handling subsystem could not be initialized
6	JFBATCH Initialization Error Program could not interpret the command line
7	JFBATCH Initialization Error Program could not initialize the trace log facility

Num	Error Codes
8	JFBATCH Initialization Error Program could not initialize the error log facility
9	JFBATCH Invalid priority value typed on the command line
10	JFBATCH Invalid registration value(s) typed on the command line
11	JFBATCH failed to add the batch to the requested task
12	JFBATCH Initialization Error Could not locate a valid FormWare base directory
13	JFBATCH Initialization Error Could not initialize properties facility
14	JFBATCH Error The HARDCOPY batch option was specified, but no BDF was created.
15	JFBATCH Error The HARDCOPY batch option was specified, but no job name was specified.
16	JFBATCH Error An invalid image count value was specified.
18	JFBATCH Error User canceled the operation.
30	JFBATCH Error Unable to access the BDF to validate the image count.

Creating Batches by Importing Images

You can import ASCII text data into a batch. The batch import requires data formatted with a two-digit form number followed by the form data, a carriage return, and a line feed. Each record must be formatted this way. To use this option, make certain the data are placed on the system in a file associated with a device.

To perform a batch import:

1. Open Administration.

2. Display the contents of the \FormWare\Files folder in the File Browser.

3. From the Process menu, select the **Batch Import** option. The Batch Import dialog displays.

4. In the **Job Name** field, type the name of the job to be associated with the batch data file created by the import process.

5. In the **Batch Name** field, type the name of the BDF and batch data file created by the import process. To save the batch in folder other than the \FormWare\Files folder, type the name of the batch and include the path. This will create both the batch file and the sub-folder.

6. In the **Device Name** field, type the name of the import device that contains the data.

7. Select **OK** to create the new batch in the \FormWare\Files folder.

Note: You can also run the program from the command line with the following: `fwedtlng –IMP:<jobname>,<import device> <path to batch file>`

Creating Batches from an XML Import

FormWare supports three XML input formats:

- If the XML file contains only image information, then output is a BDF.
- If the XML file contains only field data, then output is a batch data file (BCH) and a BDF with only a HARDCOPY=Y parameter.
- If the XML file contains both image and field information, then a BDF and BCH are output.

To import an XML file into the FormWare system and create a batch:

1. Open Administration.

2. Display the contents of the \FormWare\Files folder in the File Browser.

3. From the Process menu, select the **XML Import** option. The XML Import dialog displays.

4. In the **Job Name** field, type the name of the job to be associated with the batch data file created by the import process.

5. In the **Batch Name** field, type the name of the BDF and batch data file created by the import process. To save the batch in folder other than the \FormWare\Files folder, type the name of the batch and include the path. This will create both the batch file and the sub-folder.

6. In the **XML Name** field, type the name of the XML file to be imported.

7. Select **OK** to create the new batch in the \FormWare\Files folder.

Import is a stand-alone process, separate from a JobFlow task. For continued processing within FormWare, you must add files manually to an existing JobFlow. XML Import uses the RPD (read page data) process. Up to 4,000 characters per image can be read from a BDF.

Running a Task Outside of JobFlow (Ad Hoc)

Ad hoc processing refers to selecting one or a number of files in the Administration window and then selecting a single process to run on those files. When you process files ad hoc, the BDFs run through the selected process only and are not routed to the next task.

To run an ad hoc process:

1. Open Administration.

2. Display the contents of the \FormWare\Files folder in the File Browser.

3. Select the **Show All Files** button on the Administration window toolbar to display all the files within the Files folder in Contents area.

4. Select one or more of BDF files in the Contents area.

5. From the Process menu, select from the following options: **Image Processing**, **Completion Processing**, **Batch Processing**, or **JobFlow**.

6. This displays a dialog for selecting a specific process to run against the files. Select a process and select **OK** to run the process. A progress dialog displays for each of the processes, displaying information about the BDFs and images running through the process. A progress dialog displays regardless of whether you run JobFlows or Ad Hoc processes.

    The title bar of the progress dialog displays the name of the current process followed by the image name. A progress bar displays the time remaining to complete the process. Progress dialogs include:

    o   **BDF** - Displays the name of the BDF currently processing and the total time logged for processing the BDF.

    o   **Image** - Displays the name of the image currently processing and the time logged for processing the current image.

    o   **Count** - Reflects the number of image files in the current BDF, the order of the processing image, and the average time logged for the images in the BDF to be processed.

    o   **Component** - Displays the name of the component currently running.

7. Select the **Thumbnail** check box to display a thumbnail of the image as it is processed.

8. Select the **Pause** button to stop the current process. Select **Resume** to continue the process.

9. Select **Cancel** to discontinue the current process on the BDF and processing image.

10. Select the **Details** button to display a list of the BDF name, image name, processing time, and status for all of the images in the BDF that have already been processed. If the BDF is running through more than one component, the information is listed for each component executed. In this case, you will see images listed more than once in the Details area. Select the **Hide Details** button to hide this information.

Progress dialogs can display for the Scan, IPP, Export, and Cleanup processes. The Image Quality Check, Completion, and Rescan processes display a specific application window that requires user input instead of a progress dialog.

Running JobFlow Tasks from Administration

The JobFlow Client runs processes configured for a JobFlow on BDFs assigned to the JobFlow. JobFlow routes these BDFs from one process, or task, to another as work is completed. A JobFlow can coordinate the work for multiple workstations in a network, or it can run on a single workstation.

To add a batch to a JobFlow queue from Administration:

1. Open Administration.

2. Display the contents of the \FormWare\Files folder in the File Browser.

3. Select the **Show All Files** button to display the BDF files in the Contents area.

4. Select one or more BDFs in the Contents area and then select **JobFlow** from the Process menu. The JobFlow dialog displays.

5. In the **JobFlow System** field, select a JobFlow. The tasks for that JobFlow then display in the **Task** field.

6. In the **Task** field, select the task in the JobFlow to which you want to add batches.

7. Select the **Add** button to specify any registration numbers for the batch.

8. In the **Priority Number** field, select a priority for the batches.

9. Select **OK** to add the batches to the JobFlow.

Logging on to a JobFlow Task

Once batches are assigned to JobFlow tasks, operators can log on to a JobFlow task to perform assigned work. A network JobFlow might include several workstations dedicated to scanning, workstations conducting Recognition and Export, and a number of workstations dedicated to Completion. When running on a network, JobFlow routes BDFs from one task to the next in the JobFlow, but can also route BDFs based on an assigned error code.

If the job INI file contains the JNF=Y parameter, Completion operators can select a job before selecting a batch data file. Completion will use the settings in the job INI file to locate an alternate directory where batch data files are stored for that particular job and display the list to the operator.

Perform the following steps to log onto a JobFlow system outside of production.

1. Open Administration.

2. Select the **JobFlow Client** button on the toolbar. The JobFlow Client dialog displays.

3. In the **JobFlow** field, select a JobFlow name from the list.

4. In the **Tasks** field, select the JobFlow task you want to perform. The number of batches available for the task displays.

5. Select **OK** to log on to the workstation to the specified JobFlow and task. Only one batch is sent to the workstation at a time.

Selecting Transaction Order

To set the order in which transactions are sent to the workstation:

1. Open the JobFlow in JobFlow Designer.

2. Open the Properties dialog for a Completion, Export, or Custom task.

3. Select the Transaction Processing tab.

4. Select the **Allow Operators to Select Transactions** check box.

5. Save and close the JobFlow.

6. From Administration, select the **JobFlow Client** toolbar button and run the JobFlow with the task containing this option. When the JobFlow Client processes this task, the Select a Transaction to Process dialog displays.

7. Highlight a batch to process and select **OK**. JobFlow Client then processes the batch using the selected task.

Note: You can also log onto a JobFlow system and task using command line parameters. Set these options in a shortcut on the desktop or in a start up group to prevent the JobFlow operator from accessing JobFlow through Administration. To launch the JFClient program, specify the JobFlow name and task number. For example: `C:\FORMWARE\PROGRAMS\JFCLIENT.EXE /f=<JOBFLOW NAME> /n=<TASK NUMBER>`

Running Manual IQC

If you have defined a Manual Image Quality Check (IQC) task in your JobFlow, an operator can check rejected images. Any forms that fail Automatic IQC should be sent to the Identify Images task. When this task runs, thumbnails of the images display in the Manual IQC dialog. You can view the thumbnails and take the following actions for images that were not scanned correctly:

- Reject the image, to be addressed in the future.
- Rescan the image at that time (if a scanner is attached to the workstation).
- Modify the image at that time by:
  - o Registering the image
  - o Rotating the image
  - o Inverting the image
- Perform high-speed Manual Form ID (see Performing Manual Form ID)
- Delete the image

Note: The Manual IQC task also produces thumbnails for color images. You can set the height and width of your thumbnails in the Image Thumbnail area on the Job Properties, Image QC tab.

> **Note**: Run the IQC/Rescan mode before any of the other Completion modes.

The IQC/Rescan JobFlow displays thumbnail files of the images in a BDF. Use this task to reject, modify, or rescan the images. To conduct any of these operations on an image, you must select the field corresponding with the thumbnail to select that image. The path of the selected image displays in the status bar.

Processing Images in Manual IQC

To process images in the Manual IQC task:

1. Open a batch in Manual IQC.

> **Note**: The menu and toolbar items specific to Manual IQC display when you access Manual IQC from Administration (**Process >Completion Processing > Image QC/Rescan**) or JobFlow (**Manual Image QC/Rescan JobFlow** task). If you access Manual IQC from Completion using the **File > Image QC/Rescan** menu item, the menu items specific to Completion remain in the window.

2. Use the following toolbar buttons (or select the options from the menus) to adjust or change the image:

    o   **New File** - Opens a new BDF.

    o   **Previous Form** - Moves the cursor to the previous form in the BDF.

    o   **Next Form** - Moves the cursor to the next form in the BDF.

    o   **First Form** - Moves the cursor to the first form in the BDF.

    o   **Last Form** - Moves the cursor to the last form in the BDF.

    o   **Current Form** - Moves the cursor to current form in the BDF.

    o   **Reject Form** - Displays the Select Reject dialog for Manual IQC, used to reject an image. See the Rejecting an Image section for more information.

    o   **Previous Field** - Moves the cursor to the previous reject field in the current BDF.

    o   **Next Field** - Moves the cursor to the next reject field in the current BDF.

    o   **First Field** - Moves the cursor to the first reject field in the current BDF.

    o   **Last Field** - Moves the cursor to the last reject field in the current BDF.

    o   **Cut** - Cuts the current image. This option is used with the Paste function to reorder the images in the BDF.

    o   **Paste** - Pastes the cut image after the currently selected image. This option is used with the Cut function to reorder the images in the BDF.

    o   **Assign Form** - Displays the Assign Form dialog, used to associate a data form with the current image. Type a number for the Form Number or select a Form Name and then select **OK**.

    o   **Assign Variant** - Display the Assign Variant dialog, used to associate the data form with a particular form variant definition. See the Setting Virtual Form ID Properties section in Designing with FormWare for more information on form variants.

    o   **Rotate Image Right** - Rotates an image 90° clockwise. Select this option more than once to continue to rotate the image. The rotate functions only apply to the image when it is displayed in full screen view.

    o   **Rotate Image Left** - Rotates an image 90° counterclockwise. Select this option more than once to continue to rotate the image.

- o **Rotate Image Over** - Rotates an image 180° so that it displays upside-down from the original viewpoint.
- o **Register Image** - Registers the selected image.
- o **Invert Image** - Displays the image in negative, so the black portions of the image display as white, and the white portions of the image display as black. This function only applies to the image when it is displayed in full screen view.
- o **Full Screen Viewport** - Displays the thumbnail image in the Manual IQC window. You can apply image functions to the image when it is displayed full screen and save them to the image file. Select this option again to return the Manual IQC window to its original display.
- o **Switch Image** - Switches the current image.
- o **Image Rescan** - Scans a single piece of paper, overwrites the current image with the image generated from the scanner, and displays the new image. The new image has the same name as the original. This option is only available for a workstation attached to a scanner.
- o **Image Print** - Sends the current image to the printer.

3. In the field that displays next to each thumbnail, type a reject code if necessary. An operator might reject an image because it has the wrong orientation, is unreadable, or it might be the wrong image. To reject an image in Manual IQC, select the **Reject Form** toolbar button or select **Image > Reject** and then select a reject code from the Select Reject dialog. You can also type a unique reject reason if the reason has not been defined. The FW_IQC= parameter is written to the line above the image in the BDF with the reject reason. The BDF can then be routed to another process so the problem can be addressed.

4. Select the **Select** option from the Form menu to select a data form in the current job. Type the number of the data form and select **OK** to display that data form. You create form numbers on the Form Linkage dialog, accessed from the Job Properties dialog in Designer.

5. To stop the Manual IQC process, select one of the following options from the File menu:
- o **Close** – Closes the BDF and runs the next process.
- o **Interrupt** – Enables an operator to exit from an incomplete BDF.
- o **Exit** – Closes the Manual IQC window when the operator is finished.

Rejecting an Image

In many forms processing applications, some of the images in a given batch cannot be processed. This can be because the image is not readable, the image is not the right form type, the back of the form was scanned, etc. During the Manual IQC mode, operators can reject an image. The Select Reject dialog provides a list of reject reasons. A number of default reasons are provided, but you can also define custom reasons for the job in the Designer module.

To reject an image:

1. Open a batch in the Manual IQC task.

2. Locate a record that cannot be corrected and qualifies for rejection.

3. To reject the form, do one of the following:
- o Select the **Reject Form** toolbar button
- o Press **[F3]-[R]**.
- o Select **Image > Reject**. The Select Reject dialog displays.

4. Select a reason in the **Defined** list. If the reason you are rejecting the image does not display in the list, select the Other option in the **Defined** list and type a reason in the **Custom** field. This reason will be written to the BDF.

5. Select **OK**. The current image and record are rejected and the next image displays.

Rescanning Images During Manual IQC

If a scanner is attached to the current workstation, follow these instructions to rescan a BDF image.

1. Select the BDF image to rescan.

2. Place the paper document in the scanner.

3. Select the **Image Rescan** toolbar button in the Manual IQC window.

4. This initiates the scanner, scans the paper document, and creates a TIFF image file of the paper document in the same folder as the original image. A progress dialog displays during this process.

Applying Image Functions

During the Manual IQC process you can display the current image in a full screen. Select the options on the Images menu to rotate, register, delete or invert the image to correct it for processing.

Note: To re-register an image you must be in the full screen viewport.

To view the image in a full screen:

1. Run the Image QC/Rescan JobFlow task.

2. Select an image.

3. Select the **Full Screen Viewport** toolbar button. The image fills the screen.

4. Select the options on the Image menu to rotate, register, delete or invert the image. When you apply one of these functions, you are prompted to save the change to the original image file.

5. Select **OK.** The thumbnail and image files are updated with the change.

6. Select the **Full Screen Viewport** toolbar button a second time to return to the Manual IQC window.

Performing Manual Form ID

The Manual Form ID process typically occurs during the Identify Images mode of Completion and enables Completion operators to determine the production image that corresponds to a data form. Manual Form ID is used when the Automatic Form ID process fails to identify an image. During Manual Form ID, the Completion operator is presented with the unidentified images and types the correct form number. This form number is then written to the BDF to associate the image with the data form.

For more information on types of Form ID, see Setting Form Identification Properties in the Designing section of this Guide.

Completion operators can access Completion and perform Manual Form ID a number of different ways. The method depends on how the process is configured.

- Using Administration, an operator can open the Completion module and select a batch.
- Using a short cut on the desktop, a batch can be assigned to an operator so that the operator can double-click a shortcut to that batch on the desktop to open it in Completion.
- Using the JobFlow Client, batches can be routed to an operator.

There are also a number of different ways to perform Manual Form ID functions within a Completion session.

- Using the Identify Images (Manual ID) mode of Completion with or without a custom Manual ID menu display.
- Using the Image Quality Check/Rescan (Manual IQC) mode of Completion with or without an enhanced folder structure functionality over the Manual ID menu.
- Using other modes of Completion with the enhanced folder structure Manual ID menu.

See the Setting Manual Form ID Properties section for instructions on how to select and configure the appropriate type of Manual Form ID for your application.

Using Identify Images Mode to Perform Manual Form ID

Traditionally, Manual Form ID occurs within the Identify Images mode of Completion. You can design your application to route batches that fail Automatic Form ID to the Identify Images mode of Completion so that a Completion operator can manually identify those images.

Note: If you are running this process as a JobFlow task, it must *follow* the Image QC task for Automatic Form ID within your JobFlow.

During the Completion – Identify Images mode, a large image viewport displays a portion of the image to the Completion operator, and a small data entry screen that contains a two-character field bottom left where you type in the form number you want to assign to the image.

To perform Manual Form ID in Completion – Identify Images mode:

1. Open Administration.

2. Select the **Completion** toolbar button.

3. Close the Select a Job dialog.

4. From the File menu, select the **Identify Images** option. The Select a Job dialog displays.

5. Highlight a job in this dialog and select **OK**. The Enter New File Name dialog displays.

6. Type the name of an existing batch that corresponds to the selected job in the **File Name** field.

7. Select **OK.** The Completion window displays the first image in the batch that does not have a form number associated with it in the BDF or that has been assigned a form number of 99. A preset data form displays the name of the selected job and a single data field, and the image number displays in the far right of the status bar.
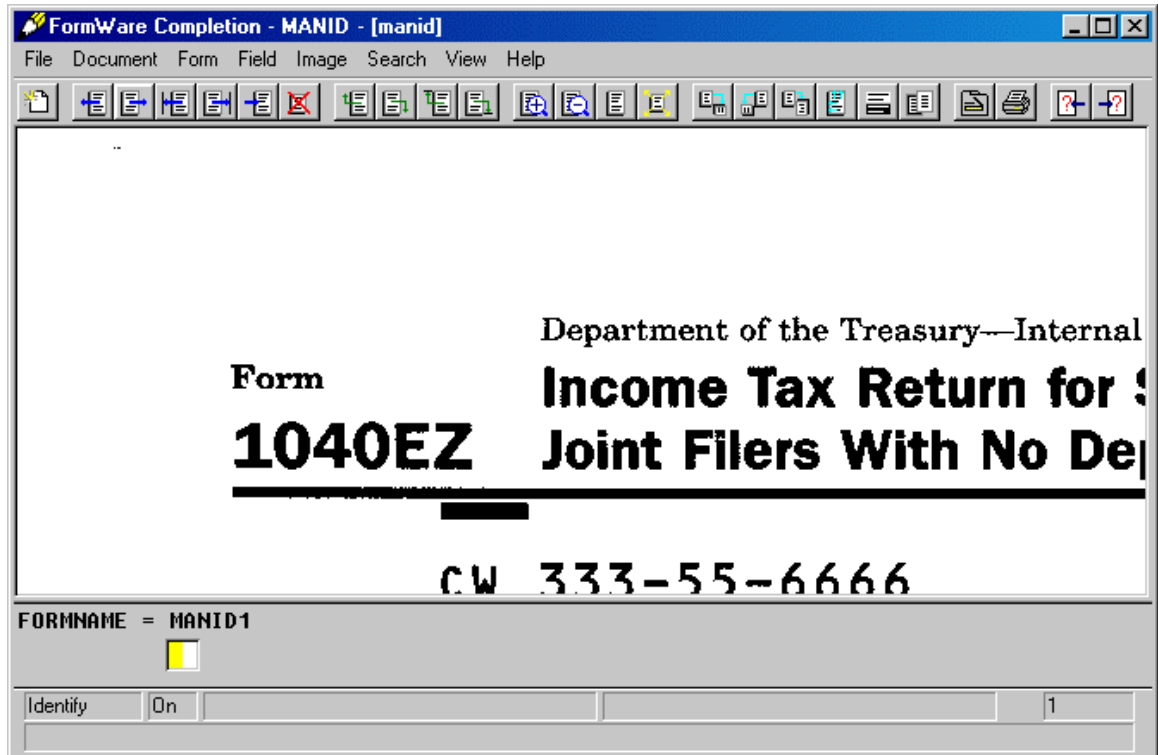
8. Type the form number in the data field and press **[Enter]**. An error message displays if the number you type does not correspond to an existing form number in the job. If the form number is correct, the next unidentified image in the batch displays.

9. Select **File > Close** when all of the images in the batch are processed.

Using a Custom Manual ID Menu in Identify Images Mode

Using the Identify Images mode of Manual Form ID, you can also incorporate attachment viewer functionality and a custom Manual Form ID menu (IDM) file. This display format presents a custom IDM file next to the image during the Completion session. Select the menu option that corresponds to the current image and the Form ID number displays for that menu option in the field next to the menu. Press [F5]-[F9] to display the Change Form dialog (IDM file), where you can select the appropriate form type from the list.

To set up an IDM file to use attachment viewer functionality and the custom menu, see the Configuring Manual ID to Use Display Option Two section within Setting Manual Form ID Properties in Designing.

Here is an illustration of this Completion mode with the split screen and custom menu.

Using Manual IQC to Perform Manual Form ID

One of the ways to perform Manual Form ID is using the Image Quality Check/Rescan (Manual IQC) mode of Completion. During the Manual IQC mode, images are presented as thumbnails and operators can review image quality and correct bad images, ensure batch integrity, and assign Form ID numbers/names to images that have failed to be identified by the Form ID module. Operators can perform basic Manual IQC tasks on each image such as rotating, rescanning, or rejecting each image. Press [F5]-[F9] to display the Assign Form dialog and assign a form type to each thumbnail image. Once a form type is assigned, the form name displays in the lower left corner of the thumbnail image.

To set up Manual Form ID to use Manual IQC in Completion, see the Configuring Manual ID to Use Display Option Three task within Setting Manual Form ID Properties in Designing.

To perform Manual Form ID in Image Quality Check/Rescan (Manual IQC) mode:

1. Open a batch in Completion -- Image Quality Check/Rescan (Manual IQC) mode to see the assigned thumbnails of records available for Manual IQC.

2. Perform the usual Manual IQC functions as defined in Running Manual IQC. As you navigate to an image that has failed Form ID, press **[F5]-[F9]** to display the Assign Form dialog.
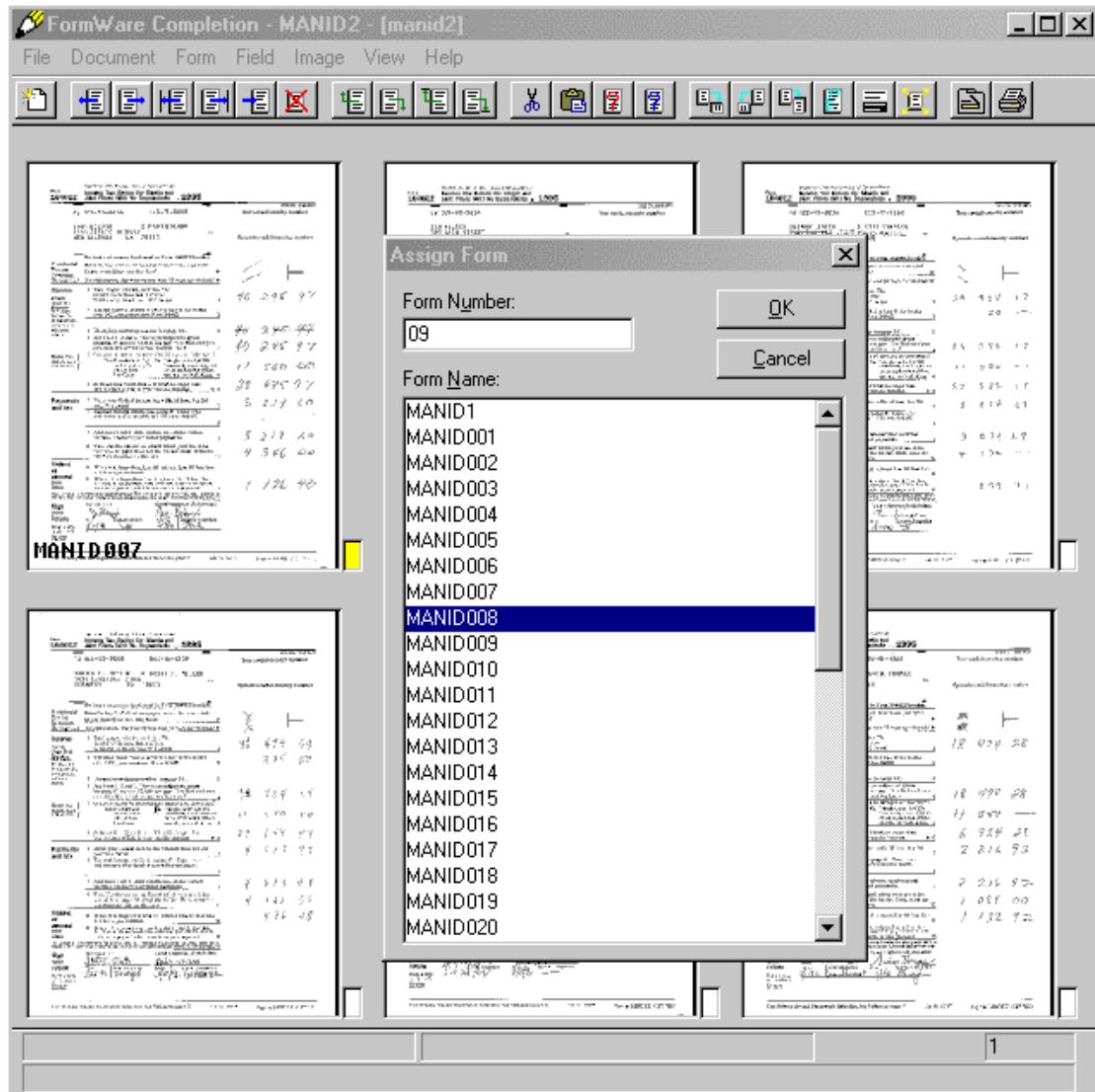
3. Select the appropriate Form Name from the list or type a valid form number in the **Form Number** field.

4. Select **OK**. The selected Form ID name displays in the lower left corner of the image thumbnail.

5. Continue until all images in the batch are processed.

6. Select **File > Close** when all of the images in the batch are processed.

Using Manual IQC to Perform High-Speed Manual Form ID

The standard method of Manual Form ID in Manual IQC (see the Using Manual IQC to Perform Manual Form ID task for more information) is best when you expect better than average recognition results and anticipate having to correct only a few rejected or unidentified forms. However, in other situations Form ID may be less successful and require the use of high-speed Manual Form ID with the Manual IQC mode.

This is appropriate when:

- Manual Form ID involves identifying many form types.
- Manual Form ID involves multiple versions of the same form coming from different sources.
- Completion operators are required to identify large amounts of records with poor recognition results and will likely encounter a large number of failed records.

To use Manual IQC for high speed Manual Form ID, you must include two command line parameters with the Completion – Image Quality Check/Rescan mode—the MID=Y parameter and the IDM=<IDM filename>.idm parameter. The MID=Y parameter forces automatic display of the IDM file over top of the image thumbnails within Completion – Image Quality Check/Rescan mode. The IDM=<idm filename>.idm parameter enables high-speed Form ID within Manual IQC.

- **-MID=Y**—a command line/client parameter switch that triggers Manual IQC to perform high-speed Manual Form ID. In -MID=Y mode, Manual IQC displays only the images that did not get assigned a Form ID. In the absence of the -MID=Y switch, Manual IQC functions as defined in Running Manual IQC.
- **-IDM=<filename.idm>**—a command line/client parameter switch that enables the association of unique IDM file with a JobFlow task. When you enable Form ID for a job, an IDM file is automatically created for that job. The job IDM file is stored in the \FormWare\Jobs folder and bears the same name as the job with an .idm extension. However, you can create and apply unique IDM files at the JobFlow task level. Inserting this parameter in the **Client Parameters** field on the JobFlow Task Properties dialog for the Manual IQC task allows you to display the Manual Form ID menu as part of the Manual IQC task.

To configure Manual IQC for performing high-speed Form ID ensure that your IDM file contains the VWR=520,0,50,0,1,0 setting to define the full image view and the setting FWN=-1,0,0,0 to define where the IDM menu will appear on the screen.

For configuration instructions, see the Configuring Manual ID to Use Display Option Four task in Setting Manual Form ID Properties section of Designing.

Once you have configured a Manual IQC task in your JobFlow, Completion operator can run this process and check rejected images. Any forms that fail Form ID are sent to the Manual IQC task. When the task runs, the images display in the Manual IQC dialog. During image processing you view unidentified images and select the correct form type.

To use Manual IQC to perform high-speed Manual Form ID:

1. Add a batch to a JobFlow. See the Adding Batches to a JobFlow task for instructions.

2. From Administration, open JobFlow Client.

3. In the **JobFlow** field, select the JobFlow that contains the Manual IQC for high-speed Manual Form ID task.

Note: The JobFlow you select must be a job in which the FormWare administrator has defined a Manual IQC task and supplied three mandatory parameters (Completion mode, -MID=Y, and -IDM=<idmfile.idm>) on the command line for Completion Manual IQC.

---

4. In the **Task Description,** select the **Image Quality Check/Rescan** task**.**

5. Select **OK** to process the batch through the selected Manual IQC task. This runs the Completion – Image Quality Check/Rescan task. Images are presented on a split screen, with a full size image on the left and image thumbnails on the right. When the Assign Form dialog is active, the display of the underlying image thumbnails is suppressed for system performance.

6. Select **Cancel** on the Assign Form dialog to close the dialog and display the image thumbnails area or press **[F5]-[F9]** to display Assign Form dialog after it is closed.

FormWare automatically generates thumbnails, and in Manual IQC, the thumbnail generator also produces viewable thumbnails for color images. Set the height and width of image thumbnails on the **Job Properties > Image QC > Image Thumbnail** area. See the Defining Image Quality Check Properties in Designing for instructions.

7. To navigate open and closed folders on the Assign Form dialog highlight the folder name and press **[Enter].**

8. To select forms on the Assign Form dialog:

    a. Highlight the form name in the folder structure and press **[Enter]** or select **OK**.

    b. Type a valid form name or form number into the **Form Number** field. Entries in the **Form Number** field are case sensitive. As you type the number or name, the cursor moves to that location on the Assign Form dialog. Select **OK** to accept the selection.

9. Press **[Enter]** to select the form name that corresponds to the image (even if you type a partial name or description into the Form Number field). Releasing that record and going to the next record will again display the Assign Form dialog. The image viewer advances through the unidentified images in the batch. Once you have completed processing all unidentified images the Assign Form dialog closes once the last image needing identification has been processed.

10. Perform Manual IQC tasks on any image in the batch by:

    a. Selecting **Cancel** on the Assign Form dialog to close it and display the image thumbnails.

    b. Using the form navigation buttons on the toolbar to move to the form you want to change.

    c. Selecting the **Full Screen Viewport** button on the toolbar to full the current image thumbnail in a full screen.

    d. Rotating, inverting, rejecting, or rescanning the current image using the toolbar buttons or menu options.

    e. Selecting **OK** on the confirmation dialog to save your changes to the image on disk.

    f. Selecting the **Full Screen Viewport** button again to toggle back to the image thumbnails.

    g. Pressing **[F5]-[F9]** to display the Assign Form dialog and continue Manual Form ID processing.

Note: While the thumbnail displays you can right click the image to see the image properties, right click the thumbnail to see the field properties, or the area outside the thumbnail to see the form properties.

11. Repeat the above steps until all images in the batch are processed or until you want to interrupt the batch.

12. Select **Cancel** to interrupt the batch and then selecting **Interrupt** from the File menu.

13. Press **Last Form** or [**F7]** to move to the end of the batch.

14. When the entire batch is processed, select **OK** at the confirmation dialog to save your changes to the batch.

15. Select **File > Close** to exit the session.

Using the Manual ID Menu with Other Completion Modes

You can use the Manual ID menu (IDM file) with modes of Completion other than Image Quality Check/Rescan (Manual IQC) or Identify Images (Manual ID). By supplying the IDM=<IDM filename>.idm parameter as a client parameter (for a JobFlow task) or on a command line with a particular Completion mode, you can enable the Change Form dialog for that Completion mode.

Because the IDM file applied here is task-specific, you can create unique Manual ID menus (IDM files) for all of your Completion mode tasks if you wish.

Follow these steps to use the Manual ID menu with a Completion mode task.

1. Create a JobFlow with a Completion mode task and follow the steps to configure that task to use a Manual ID menu. See Configuring Manual ID to Use Display Option Six task in the Setting Manual Form ID Properties section of Designing for instructions.

2. Add a batch to a JobFlow. See the Adding Batches to a JobFlow section for instructions.

3. From Administration, open JobFlow Client.

4. In the **JobFlow** field, select the JobFlow that contains the Completion mode task with the Manual ID menu.

Note: The JobFlow you select must be a job containing the Completion task with the Manual ID menu and the mandatory parameter, -IDM=<idmfile.idm> as a client parameter.

5. In the **Task Description,** select the Completion mode task.

6. Select **OK** to process the batch through the Completion mode task. This runs specified task in Completion.

7. Press **[F5]-[F9]** to display the Change Form dialog. The following illustration shows the Change Form dialog displayed in a Completion – New mode task:
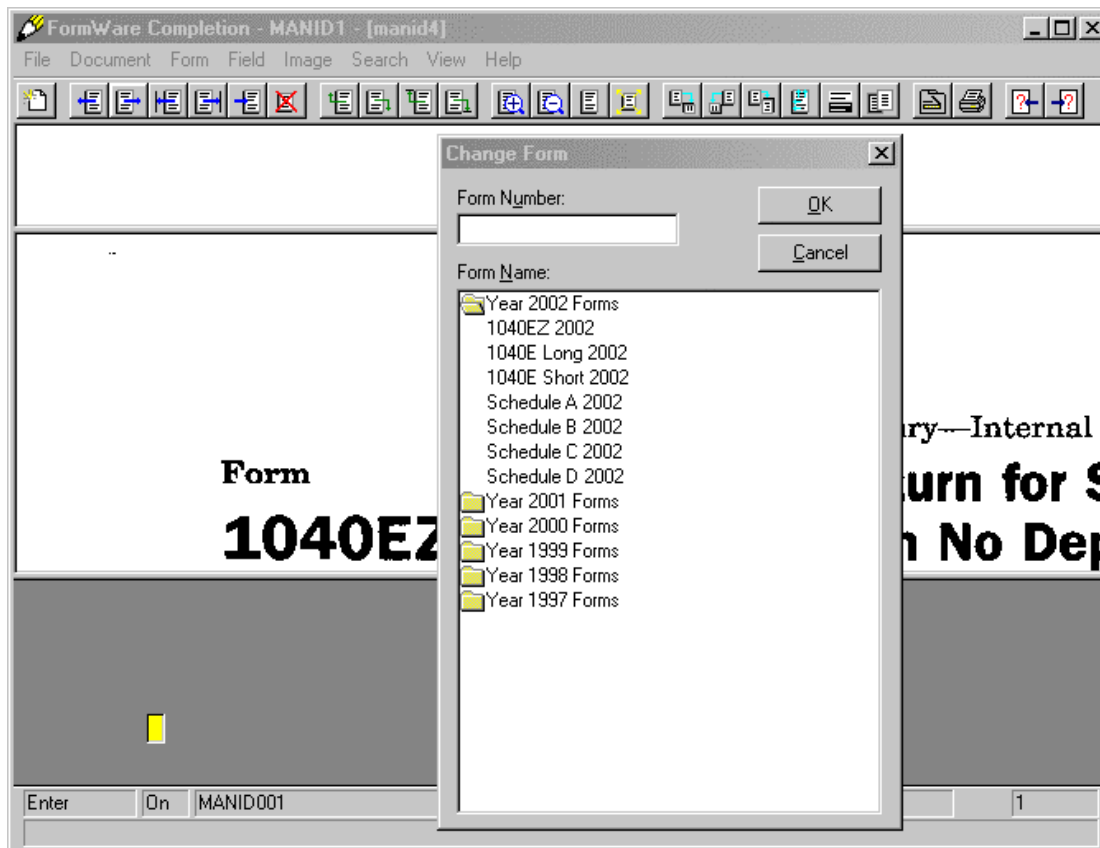
8. To navigate open and closed *folders* on the Change Form dialog, highlight the folder name and press **Enter.**

9. To select a *form* on the Change Form dialog:

   a. Highlight the form name in the folder structure and press **Enter** or select **OK**.

   b. Type a valid form name or form number into the **Form Number** field. As you type the number or name, the cursor moves to that location on the Change Form dialog. Select **OK** to accept the selection.

Note: Entries in the **Form Number** field are case sensitive.

10. To perform standard tasks related to this Completion mode:

   a. Select **Cancel** on the Change Form dialog to close it and return to Completion.

   b. Use the toolbar buttons and menu options available on the Completion window to process the batch. Options available will depend on the mode of Completion.

   c. Select **OK** on the confirmation dialog to save your changes to the image on disk.

   d. Press **[F5]-[F9]** to redisplay the Change Form dialog and perform additional Manual Form ID processing.

11. Repeat the above steps until all images in the batch are processed or until you want to interrupt the batch.

12. Select **Cancel** to interrupt the batch and then selecting **File > Interrupt**.

13. Select **Last Form** or press **[F7]** to move to the end of the batch.

14. When the entire batch is processed, select **OK** at the confirmation dialog to save your changes to the batch.

15. Select **File > Close** to exit the session.

Running Recognition

FormWare runs a separate recognition process for each of the recognition engines. You can run a recognition process in an ad hoc fashion from the Administration module or through JobFlow.

Running Recognition as an Ad Hoc Process

Ad hoc processing refers to selecting different BDFs to run a single process from within the Administration module. With ad hoc recognition processing, properties pertaining to the JobFlow component or process are defined in the job. Those properties are referenced in each of the selected BDFs.

Follow these steps to perform ad hoc Recognition:

1. Open Administration.

2. Display the contents of the \FormWare\Files folder in the File Browser.

3. Select the **Show All Files** button to display the BDFs in the Files folder.

4. Select one or more BDF files in the Contents area.

5. From the Process menu, select the **Image Processing** option. The Image Processing dialog displays.

6. Select the **Recognition** option.

7. In the Engine field, select a recognition engine.

8. Select **OK** to initiate the Recognition process against the selected BDFs. A progress dialog displays information pertaining to the recognition process.

Running Recognition from the JobFlow

JobFlow enables you to run a number of different processes on BDFs. To perform Recognition within a JobFlow, you must define at least one Recognition task in that JobFlow.

Follow these steps to run Recognition within a JobFlow. These steps are an example of the procedure you use for testing Recognition in a JobFlow, but are not completely indicative of a production environment.

1. Open Administration.

2. Display the contents of the \FormWare\Files folder in the File Browser.

3. Select the **Show All Files** button to display the BDFs in the Files folder.

4. Select one or more BDF files in the Contents area.

5. From the Process menu, select the **JobFlow** option. The JobFlow dialog displays.

6. In the **JobFlow System** field, select a JobFlow file.

7. In the **Task** field, select the Recognition task. You can also assign the BDF to a task preceding the Recognition task.

8. Select **OK** to assign the BDFs to the Recognition JobFlow task.

9. From the Administration window, select the **JobFlow Client** toolbar button. The JobFlow Client dialog displays.

10. In the **JobFlow** field, select the name of the JobFlow. The tasks in the JobFlow display in the Tasks list.

11. Select a Recognition task from the task list, or the task preceding Recognition to which you assigned the BDF.

12. Select **OK.** The first available batch begins processing. A progress dialog displays during Recognition processing.

Performing Completion

A batch is a collection of pieces of paper that you want to process together. A batch can be grouped by any user-defined criteria – a specific number limit (e.g., always 50), geographical location, document type, etc.

The Batch Definition file (BDF) is a text file that contains the job name and a list of images associated with a batch. The BDF determines the order in which the images are processed. The batch data file contains the actual data or information captured from a batch of paper forms by recognition technology and manual data entry in the Completion module. The batch data file is a binary file that can only be modified by FormWare modules and programs.

Batch data is generated in the Completion module by performing data entry from paper documents, from images, or by post-recognition reject repair. When an operator performs post-recognition reject repair on a batch, an out-of-context reject repair Completion window displays.

Batch processing is not complete until all rejects, images, or paper documents are processed. When batch processing is complete, the operator closes it and the batch status is then set to Terminated. To process another batch, the operator logs on to JobFlow and is presented with another batch.

By default, if the operator does not process all the images in a batch (in reject repair or key from image jobs), the batch cannot be closed. Closing a batch before it is complete displays an error message to the operator. An operator can choose to exit from an incomplete batch by selecting the Interrupt option. The Interrupted status enables a supervisor to determine the batches that have not been completed.

Batches can be closed, interrupted, or verified depending on whether all records in the batch have been processed. Use the Administration module to check the status of any batch.

After you start the Completion module, you should become familiar with the Completion module window. The Completion module uses a familiar Microsoft Windows interface and is divided into several parts. The Completion window is divided into the following areas:

- **Title Bar** – The title bar at the top of the window displays the current job and batch names.

- **Reject Ticker Tape**– The reject ticker tape in the top portion of the window displays out-of-context, questionable or low-confidence characters during a Reject Repair session. The current reject character is highlighted in this window, as well as its location in the image viewport and the data field. When a recognition engine returns a value, it also returns a number (usually between 0 and 100). This number is a percentage of how confident the engine is about the value returned. This percentage is called the confidence level. The characters in the ticker tape are ones that did not meet the confidence level specified for each field within the Designer module. The ticker tape can contain characters from just one data form or for an entire batch, depending on the parameters set for the job.

- **Image Viewport** – The image viewport in the middle of the window displays the image. The recognition zone and current reject character for the field are highlighted on the image. You define the highlight colors in the Designer module.

- **Image Ruler** – Completion operators use the image ruler to track their place in a table or column while typing information. It displays as a red line above the row that the operator is currently keying. The image ruler is available only inside the Completion window.

- **Data Form**- The data form displays at the bottom of the window at it was set up in Designer. The data form contains the data fields used to collect the data you want to extract from an image or paper form. It typically replicates the arrangement of fields and labels on the paper form. Recognition data displays in the fields and reject characters display as question marks (?) with a yellow highlight. The cursor moves from reject character to reject character as the operator types in the correct characters. The operator can repair questionable characters or retype fields that contain too many errors.

- **Status Bar** - The status bar at the bottom of the window displays information about the current data form, field, and record as well as the Completion session. When using the Attachment Viewer, the number of attachments displays in the status bar between the field name and the record number.

The completion keyboard is either Standard or 029. The 029 keyboard allows only uppercase letters and provides a numeric keypad under the right hand.

Viewing Images in Completion

When a Completion operator receives a batch, the Completion window displays with many options for correcting data and moving among fields and records within the batch. Image functions enable the operator to manipulate the image displayed in the image viewport. You can use the toolbar on the Completion window or select options from the Image menu.

To manipulate an image:

1. Open a batch in Completion and view the image in the image viewport.

2. Use the following options to manipulate the image:
    - **Previous** - Displays the previous image in the batch.
    - **Next** - Displays the next image in the batch.
    - **First** - Displays the first image in the batch
    - **Last** - Displays the last image in the batch
    - **Current** - Displays the image and associated data form at the furthest point where recognition or reject repair has been performed.

- o **Zoom +** - Enlarges the image. Select this option a number of times to continue to increase the image size.
- o **Zoom -** - Shrinks the image. Select this option a number of times to continue to decrease the image size.
- o **Reset** - Returns the image display to the original image size after using either the **Zoom +** or **Zoom –** options.
- o **Manual Adjust** - Freezes the image in the viewport. This allows the operator to view an area of the image without it moving when a new field is selected. The image is released when the operator selects mapped key 378 or releases the record. **Manual Adjust** can be used with the image ruler.
- o **Full Screen** - Expands the Image Viewer to the entire Completion window. The reject ticker tape and data form do not display. Select this option again to return the Completion window to its original display.
- o **Rotate Right** - Rotates the image 90° clockwise. Select this option a number of times to continue to rotate the image.

Note: Rotating the image in the viewport enables the operator to type data in the form, even if the image is scanned incorrectly. Select any of the image rotation buttons to cause the image to be rotated in memory. This does not affect the image that is stored on the disk. With the rotation functions, the image remains rotated for the entire record. When the next image is displayed, the default Image Viewport functions are restored.

- o **Rotate Left** - Rotates the image 90° counterclockwise. Select this option a number of times to continue to rotate the image.
- o **Rotate Over** - Rotates the image 180° so that it displays upside-down from the original viewpoint.
- o **Register** - Registers the image so that the 0,0 point is aligned with the first instance of black data on the image. Continue to select this option to create a new registration of the image.
- o **Invert** - Displays the image in negative so that the black and white portions of the image are reversed. This changes the colors of the image from black text on a white background to white text on a black background. The image remains inverted until the next image is displayed.
- o **Switch** - Switches between displaying the original image and the enhanced image. If the original image is currently displayed, selecting this option displays the enhanced image. An error message displays if an enhanced image does not exist.
- o **Load New Image** - Loads the next image in the batch.
- o **Rescan** - Scans a single paper document, overwrites the current image, and then displays the new image. The new image has the same name and location as the original. This option is only available for a workstation with an attached scanner. You must set the SCN=Y INI parameter in either the job INI or System.ini file for this option to be enabled.
- o **Print** - Sends the current image to the printer. You must set the PRT=Y INI parameter in either the job INI or System.ini file for this option to be enabled.
- o **Properties** - Displays the Image Properties dialog with the image name, location, and dimensions. You can also access the image properties by right-clicking in the viewport. To display information about the entire batch, select **Properties** from the File menu. The File Properties dialog displays the number of records, images, errors, and forms within the batch.

Selecting a New Image

To view a new image in the image viewport:

1. Open a batch in Completion.

2. From the Image menu, select the **Select** option. The Select Image dialog displays.

3. Either type a number in the **Image Number** field or select a name in the **Image Name** field.

4. Select **OK**. The image displays with its associated data form and any data that may be in the data fields.

Resizing an Image

When viewing an image, an operator can zoom in or out, or display the full image on the screen. The zooming factor is 25%. The area in the center of the viewport is enlarged. The zooming function is active only while the cursor is on the current field. When the operator moves to the next field, the original image display is restored.

To resize the image:

1. With the mouse, click on the upper left-hand corner of the portion of the image you wish to zoom.

2. Drag the mouse diagonally to the lower right-hand corner of the zoom region.

3. Release the mouse and the selected region is enlarged.

4. To return to the regular image display, select the **Reset Image Viewport** button.

Viewing Batch Information

Options on the View menu in the Completion module allow you to view additional information about the batch other than data in specific fields. To view additional data:

1. Open a batch in Completion.

2. From the View menu, select from the following options:

   o **Accumulators** – Displays the Batch Accumulators dialog. Accumulators are 16 mathematical storage areas that are created with the batch. Accumulators display if an operator exits from a batch that is out of balance (not equal to 0). You define accumulators for the job in Designer.

   o **File Log** - A batch log is a text file created by the Field Edit or File End Edit program. You assign these programs in Designer. Batch file logs reside in the \FormWare\Files folder with a .log extension.

   o **Statistics** - Displays the Operator Session Statistics dialog that contains a summary of the current batch statistics. See the Viewing Operator Statistics section for more information.

   o **BDF Data** - Displays the contents of the current BDF file.

   o **OCR Data** - Displays the recognition data returned by the recognition engines for the current batch.

   o **Attachments** – Displays any attachments in the Attachment Viewer. See the Viewing Attachments section for more information.

   o **Ruler** – Displays the image ruler in the image viewport. See the Using the Image Ruler section for more information.

Using the Image Ruler

The image ruler is intended for use primarily during Key from Image processes. The ruler works best when used on columns with equally sized rows. If the row sizes are uneven, the placement of the ruler will not be accurate as it moves down the column.

To use the image ruler:

1. Open a batch in the Completion window.

2. To activate the Image Ruler, select the **Ruler** option from the View menu in the Completion window. You can also map a keyboard key to a value of 523 and press that key while in Completion.

3. To designate where the rule will display in the image viewport:

4. Place your mouse pointer on the top left-hand corner of the first row of the column or table where you want the image ruler to display and click once.

5. Place your mouse pointer on the bottom right-hand corner of the first row in the column or table where you want the image ruler to display and click once.

   The image ruler displays as a red line above the first row immediately after the second click. The image ruler will not display if you drag after clicking on one of the corners of the row.

6. Press the [Enter] key to move the image ruler down the column as you input information.

Note: An enhancement was added to Completion to change the behavior of the Ruler option so that In a Key from Image application, the ruler continues to scroll down the image when the image viewport changes. This enables you to use the ruler on large tables that cannot display in one image viewport setting.

   The image ruler turns off when you switch from one record to the next. You must reactivate it each time you move to a new record.

7. Image functions such as zoom in, zoom out and rotate are suspended while the image ruler is active. Deactivate the image rule to use these functions.

Using Keyboard Equivalents

When operators perform high speed recognition reject repair, it is faster for them to press keys on the keyboard rather than use the Completion toolbar buttons or menus. For this reason, the keys on the keyboard are mapped so they can perform toolbar functions. The default keyboard functions are listed below. These values do not apply when you enable keyboard mapping.

Function Name	Key(s)	Description
Down Arrow	[↓]	Moves the cursor to the next field.
Left Arrow	[←]	Moves the cursor one position to the left of the screen.
Right Arrow	[→]	Moves the cursor one position to the right of the screen.
Up Arrow	[]	Moves the cursor to the previous field.
Aux Dup table	[Home]	Displays the Aux Dup table.
Word Left	[Ctrl] + [←]	Moves the cursor one word to the left.
Word Right	[Ctrl] + [→]	Moves the cursor one word to the right.
Beginning of Field	[Ctrl] + [Home]	Moves the cursor to the beginning of the field.
End of Field	[Ctrl] + [End]	Moves the cursor to the end of the field.

Function Name	Key(s)	Description
Delete	[Del]	Deletes the character at the cursor position.
Move to Tab Field	[Tab]	Moves the cursor to the next tab field.
Auto	[F2]	Turns Completion auto functions on and off.
Record Forward	[F3] [→] or [PgDn]	Positions the cursor on the first field of the next record.
Record Back	[F3] [←] or [PgUp]	Positions the cursor on the first field of the previous record.
Close	[F3] [A]	Closes the current Completion session.
Interrupt	[F3] [B]	Interrupts the current Completion session.
Go To First Record	[F3] [G]	Goes to the first record in the batch.
Go To Last Record	[F3] [H]	Goes to the last record in the batch.
Go To Record	[F3] [I]	Goes to a specified record.
Field Content Search	[F3] [J]	Enables the operator to perform a field content search.
Search for Error Flags	[F3] [L]	Enables the operator to search for error flags.
Repeat Last Search	[F3] [N]	Enables the operator to repeat the last search.
Display Statistics	[F3] [P]	Displays the current session statistics.
Record Reject	[F3] [R]	Displays the menu if the reject menu exists.
Session Suspend	[F3] [S]	Suspends the current Completion session.
Record Display	[F3] [W]	Displays all record data in Verify mode.
Record Correct	[F3] [F9]	Clears all captured field data.
Record Delete	[F3] [Del]	Deletes the current record. This option is not available with image-based applications unless the Delete Image Records option is enabled.
Record Dupout	[F3] [End]	Duplicates the remainder of the record from the previous record.
Form Help	[F3] [Esc]	Displays form help.
Record Insert	[F3] [Ins]	Inserts a record.
Field Forward	[F4] [→] or [F4] [↓]	Positions the cursor on the next field.
Field Back	[F4] [←] or [F4] []	Moves the cursor to the previous field.
Field Delete	[F4] [Del]	Deletes the data in the field and positions the cursor at the beginning of the field.
Field Flag	[F4] [F]	Places an error flag in the current position and does not release the field. Enables multiple error flags per field.
Field Display	[F4] [W]	Display the field data in Verify mode.
Field Dup	[F6]	Duplicates the data in the same field number of the previous record.
Field Correct	[F4] [F9]	Enables Field Correct mode.
Field Goto	[F4] [G]	Displays the Field Goto dialog.
Field Override	[F4] [O]	Sets the field override condition so that an operator can bypass all check box and field edits.
Job Help	[F5] [Esc]	Displays job help.
Form Dupout	[F5] [End]	Duplicates the data at the cursor position to the end of the record from the previous record of the same form.
Form Dup	[F5] [F6]	Duplicates the same field from the previous common form level into this field.
Form Change	[F5] [F9]	Enables the operator to change the current data form.

Function Name	Key(s)	Description
Dup	[F6]	Copies the contents of the field on the previous record into the current field.
Location Return	[F7]	Positions the cursor on the next keying position within the file.
Oversign	[F8]	Used to enter negative numbers. Converts the last character in the field to a letter. This letter indicates the field is negative.
Character Correct	[F9]	Causes character correct condition.
Insert	[Ins]	Changes the Data Entry mode from typeover to insert.
Page Down	[PgDn]	Moves the cursor to the next record.
Page Up	[PgUp]	Moves the cursor to the previous record.
Shift	[Shift]	Enters appropriate keys determined by the field type or keyboard mapping.
Field Accept	[Shift] [Enter]	Accepts the data currently in the field and allows edits to run on the field.

Accepting Data

As operators verify data in Completion, they may need to accept field data and ensure that no other processes affect that data (i.e., the data will not pass the edits and needs to be forced through). This is done with the field override or the field accept key. The operator must press the field override key, [F4]-[O], when the cursor is in a data field before typing any data. The field override key does not accept recognition data. It ensures that the data typed in the field will not be changed by an edit program. If the operator presses the field accept key, [Shift]-[Enter], the data in the field is accepted and edits are still run against the field.

Rejecting Data

As data is validated through the recognition and validation processes, it is sometimes necessary to reject a field or a record. Rejecting data in a field or record can either destroy or maintain the data; the default is to maintain the data. Field and record reject codes are stored in the field data so they can be detected by the FormWare JobFlow system.

If a record is rejected during Completion, the JobFlow system receives a default reject code called Exit Rejected. It has the same behavior as an Exit Success code, but with a value of 11. The batch is released from its task in Completion and can be differentiated from a batch that exited Completion successfully. This means that the batch with the rejected record can be routed to an alternative queue for exception processing.

To reject a record during Completion:

1. Select the Reject option from the Form menu. The Select Reject dialog displays a list of reasons to reject this record.

2. To select a reason, highlight the reason in the list and select **OK** to accept it and return to the Completion session.

Performing a Non-Destructive Record Reject

Non-destructive record reject is the default behavior in Completion, and does not result in the loss of any data. Instead, a flag is associated with a rejected field and is used to reference the field later. If the record has been rejected, a reject code representing the reason for rejection is also stored with the field. A red box displays around a rejected field when viewed in any of the Completion modes. The field data remains intact.

For example, if a record is rejected with non-destructive record reject enabled, the operator is presented with the Reject Selection dialog and can then select a reason for rejecting the record. This reason is stored as a code with the record and each of the fields in the record are flagged. When that record is viewed in another Completion session, the rejected fields are outlined in red.

Performing a Destructive Record Reject

Destructive record reject stores reject codes within the field data, resulting in the loss of data. With this process, a pound character (#) is placed in the first character position of the data field rejected. When a record is rejected, a reject code is also placed in the second character position of each data field within the rejected record. The placement of these reject characters destroys the existing data in those positions in the data field. You enable this destructive record reject method by adding the DEF=Y INI parameter to the job INI file.

For example, if destructive record reject is enabled, a field with "SCRANTON" data will display "#BRANTON". The pound character (#) is the reject while the B character is the record reject code.

Returning to the Location Field

When an operator is keying a record, the Completion module keeps track of the location field, location record, current field, and current record. The location record is the furthest record that the operator has reached in the batch. The location field is the field within the location record that is the next to be keyed. If an operator backs up in a batch, the current record and location record are no longer the same. By pressing [F7] (the location return key), the operator automatically advances to the location field, which is the furthest point that has been keyed in the batch.

Working with Documents

A document is a multi-page form that you want to group together for forms processing purposes (e.g., a tax form and its related schedule forms). Within each document, there is a primary form called the Document Indicator form. Each time the Completion module encounters an image of this form type within a batch, it increments the document count by one.

Generally, one or more documents can exist within a single batch, and one or more forms can exist within a single document. However, the use of documents is optional and you can alternately associate forms directly to a batch. The total number of documents repaired or typed is stored as part of the operator statistics record.

Use the following functions on the Document menu to move quickly through documents in the batch. If the application does not contain documents, all document features are disabled.

- **Previous** - Moves the cursor to the previous document, placing it at the Document Indicator record.
- **Next** - Moves the cursor to the next document, placing it at Document Indicator record.

- **First** - Moves the cursor to the first Document Indicator record in the batch.
- **Last** - Moves the cursor to the last Document Indicator record in the batch.
- **Specify** - Specifies a document number.
- **Insert** - Inserts a document with all of the data forms that make up the document. This option inserts the new document before the current document.

Note: The fields on the inserted forms are not location fields. Therefore, any field edit programs that use the if conditional "@ISLOCFLD" or the LINK verb are not executed on these forms.

- **Delete** - Deletes all records associated with the current document.
- **Properties** - Displays the Document Properties dialog.

Working with Forms

A form contains a group of fields into which data is entered. A form can be associated with an image or a portion of the image. Form functions move the cursor from form to form or to a specified form. Select the following options from the Form menu.

- **Previous** - Moves the cursor to the first position of field one in the current form, if the cursor is not already in that position. Select it a second time to move the cursor to the first position of field one in the previous form.
- **Next** - Moves the cursor to the first position of field one in the next form.
- **First** - Moves the cursor to the first position of field one in the first form of the current batch.
- **Last** - Moves the cursor to the first position of field one in the last form of the current batch.
- **Select** – Opens the Select Form dialog where you can type the number of the form you want to display.
- **Current** – Moves the cursor to the farthest point where data is typed or reject characters are repaired.
- **Change Form** – Displays the Change Form dialog, used to change the data form for the image to another form within the job. You can select the form number or form name. Typing a form number places a new form ID number in the batch and removes the number that was initially associated with the image. This option is available only when more than one data form exists in the job.
- **Insert** – Displays the Form Insert dialog used to insert a data form. An operator can type the number of a data form, which will replace the current form. Selecting the **Insert After** check box keeps the current data form and adds the new form as the next record.
- **Delete** – Deletes the current record from the batch.
- **Reject** – Displays the Select Reject dialog. From this dialog, an operator can select a reason for rejecting a record. The operator may skip the record and move to the next.
- **Properties** – Displays the Form Properties dialog that contains information such as form name, form number, number of fields, and record size. The Form Properties dialog is also visible by right clicking on the data form in the Completion window.

Working with Fields

The data field is the smallest unit on the data form into which data is placed. Field functions move the cursor between fields. The operator can also place the cursor on any field in the record using the mouse. Click on any field to position the cursor on the selected field, as long as the field has been repaired or has had data in it. The operator cannot position the cursor past the farthest access point in the batch.

Select the following options from the Field menu:

- **Previous** - Moves the cursor to the previous data field.
- **Next** - Moves the cursor to the next data field.
- **First** - Moves the cursor to the first data field in the record.
- **Last** - Moves the cursor to the last data field in the record.
- **Select** - Displays the Select Field dialog, which enables an operator to select the data field where the cursor is placed. The operator can either type a number in the **Field Number** field or highlight a field in the **Field Name** field and select the **OK** button.
- **Current** - Moves the cursor to the furthest point where recognition or reject repair has been performed.
- **Delete** - Deletes the contents of the current field and positions the cursor in the first position of this field.
- **Flag** - Places a field error flag in the field and positions the cursor on the next field. The error flag is used as a reference for the field during Remove Flags mode.
- **Properties** - Displays the Field Properties dialog for the current field. The field properties are also available by right clicking on the field. The properties that display depend upon the settings for the field made in the Designer module.

Using the Field Override Option

Use the Field Override option ([F4-O]) to bypass the checkbox edits and either an FPL After Field event or a VBA OnFieldEnd event for a particular field. You set checkbox edits for a field within Designer on the Field Properties, Checkbox edits tab, or for multiple fields within Designer on the Multi-Field Properties, Edits tab.

Accessing Help

If help files were created in Designer, the operator can use the keyboard keys to view the help. You can create help for the job, form, and field. Field-level help displays in the status bar of the Completion window.

Performing Searches

Several types of searches are available in Completion. Access a specific search function by selecting from the following options:

- **Field Content** – Displays the Field Content Search dialog. An operator can type a search text string in the **Search For** field to locate that text within fields in the batch.
- **Character Sequence** – Displays the Character Sequence Search dialog used to search for a sequence of characters in the batch. The sequence can be in one field or span multiple fields.

- **Error Flag** – Searches for error flags in the batch.
- **Unverified Form** – Searches the batch for unverified records.
- **Rejected Form** – Displays the first rejected record found in the current batch.
- **Repeat Last Search** – Repeats the last executed search.

Using the Scale to Gray Feature

Use the Scale to Gray feature to prevent image degradation that is caused by zooming in to enlarge a bi-tonal image. The larger you make the image, the more blurred it can become.

By using Scale to Gray, you can zoom in (or scale) an image to a significant degree without compromising the quality of the image. To apply the gray scale functionality, you must set the STG=Y parameter, either in the System.ini file or the job INI file.

The Scale to Gray feature is available only in the Completion module. This feature can also be set by marking the Scale to Gray checkbox on the Job Properties, Image tab in Designer.

Performing Transaction-Based Processing

Transaction-based processing typically involves processing a single image at a time as opposed to a batch of multiple images. However, the true definition of transaction processing is running the OCX version of components in a workflow. In this case, all components remain open and waiting to process work, as opposed to closing each time a batch has completed processing.

When the Completion module is in Transaction mode, it remains loaded and waits for a BDF to become available via the routing properties set in the JobFlow. When an operator finishes a batch, Completion exits and then reopens when another batch is ready. This prevents the operator from using the Completion menus between batches.

To use transaction-based processing, select the **Transaction** option from the File menu.

Adding an Oversign Character

The oversign key [F8] is a data entry standard that marks numeric fields as negative values without placing a "-" in the field data. This is useful when exporting data to a mainframe, since most mainframes cannot handle a "-" character in the data stream. You can press the oversign key before or after the last digit is entered in the field. The last digit is then translated into an alphabetic character.

For example, if the operator needs to enter the value "-3657," she can key type either:

```
3 6 5 [F8] 7     for oversign before
```

Or,

```
3 6 5 7 [F8]     for oversign after
```

Viewing Attachments

Attachments are images that are used during a Completion session, but not for extracting data. They do not have a record associated with them. Instead, they contain supplementary data that is not extracted using a recognition engine.

The Attachment Viewer displays on the left side of the Completion window. The status bar shows the total number of attachments and the number of the current attachment. You define the

Attachment Viewer settings including keyboard mapping in Designer. See the Setting Up the Attachment Viewer section.

If you selected **Include Current Image** in Designer, the first attachment is the same as the image in the Completion viewport and becomes part of the total attachment count. In normal processing modes, the Attachment Viewer will look for the attachments after the current form. In document processing mode, the Attachment Viewer will skip over data images until it locates attachments or the next document indicator. The assumption is that in document mode, each document will contain the list of images with data images appearing first followed by attachment images.

Viewing Attachment Forms in Completion

Attachment images display either when the operator chooses a menu item or automatically whenever an attachment is encountered. The attachment images display in the Image Viewer, which has **Forward** and **Back** buttons. These buttons enable the operator to scroll through the attachment images for the current record.

During a Completion session, the operator can select the **Attachments** option from the View menu to display the attachment images for the current record.

To automatically display attachments for each record, add the DAF=Y parameter to the System.ini or job INI file. Attachment images are determined by having been identified as the attachment form type and being listed after the current record image but before the next record image in the BDF. The Image Viewer displays on the screen when an attachment image is encountered. Viewing attachment images automatically assumes that Form ID has been previously run and has properly identified the data form types.

Skipping Attachment Forms in Completion

You can specify that attachment images are skipped, or not displayed, during a Completion session. Skipping an attachment image means that the image is not decompressed and is not displayed to the operator. This can save processing time and help prevent the degradation of throughput. To skip attachment forms in Completion, add the SAF=Y parameter to the job INI file. Then when an image is encountered in a BDF that has a form number corresponding to an attachment data form, the image is not decompressed and a record is not created.

Exiting a Batch

There are three ways to exit from batch within Completion. Select from the following options:

- Select **File > Interrupt** to exit from an incomplete batch. This changes the status of the batch to Incomplete. This status is helpful to Completion supervisors monitoring batch progress. In imaging applications, operators must use this option to exit from any batch when not all images in the batch are processed.

- Select **File >Close** to close a batch and remain in the Completion module. During reject repair or key from image, batches can only be closed after all images are processed (unless you have selected the Close at Anytime option for the job on the Job Properties, Options tab in Designer). In a key from paper application, a batch can be closed at any time.

- Select **File >Exit** to close the Completion module when a batch is finished. The operator receives an error message if not all the images in the batch are processed.

Running a Document Assembly Application

Once you have designed and tested your Document Assembly application, you are ready to run it in a production environment.

Tasks in this section assume that you have completed the tasks within Configuring a Document Assembly Application in Designing and, in particular, have created a JobFlow within JobFlow Designer for the Document Assembly application.

Follow these steps to run a Document Assembly application as a JobFlow task:

1. From Administration, open the \FormWare\Files folder and select **Show All Files** on the toolbar to see the contents of this folder.

2. Highlight the batch (BDF file) you want to process in the Contents area.

3. Select **Process > JobFlow** to display the JobFlow dialog.
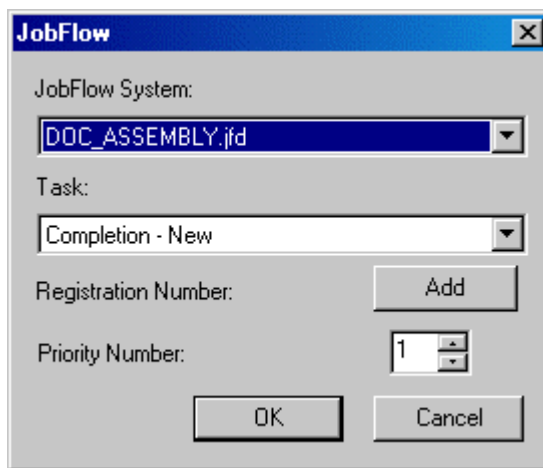
4. In the **JobFlow System** field, select the Document Assembly JobFlow in the drop-down list.

5. In the **Task** field, select Completion – New.

6. Select **OK** to add the batch to this JobFlow task. A dialog will display to confirm that the batch was successfully added.

7. On the Administration window, select **JobFlow Client** on the toolbar.
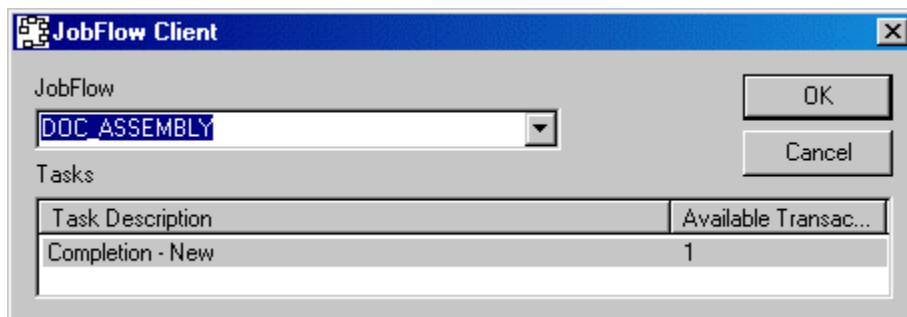
8. On the JobFlow Client dialog, select the Document Assembly JobFlow from the JobFlow drop-down list, the Completion – New task in the Task Description column, and then select OK. This runs the selected transaction in Completion. The Data Items dialog initially displays over the entire data form area of the Completion window.

Navigating the Data Items Dialog

When you first access the Document Assembly Completion session, the Data Items dialog displays automatically over the data form area of the Completion window.
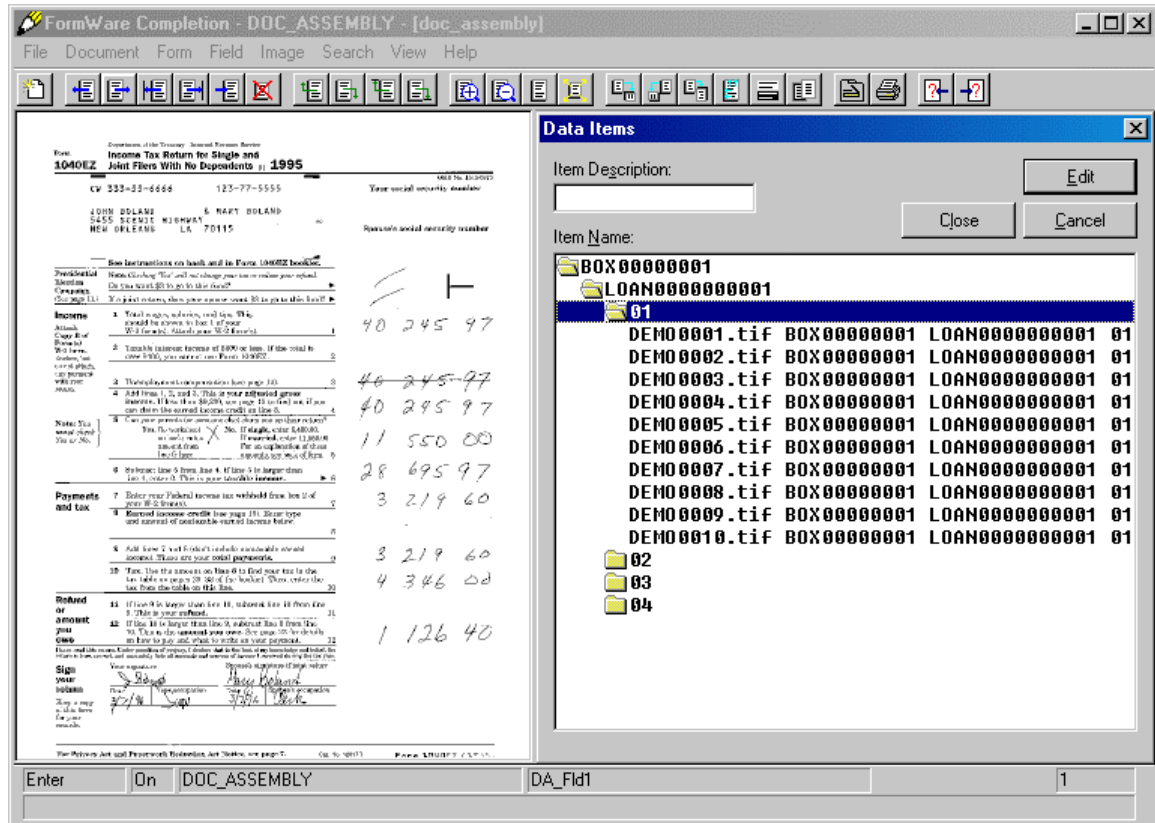
While the Data Items dialog is active, the main Completion window (and all toolbar and menu functions on it) remains inactive. Active control remains with the dialog until you select Cancel and close the dialog. Once you close the dialog, control is given back to the Completion window and the functions on the window reactivate.

The Data Item dialog displays a tree view of the data hierarchy passed to Completion for processing. The hierarchy is built based on the level tags (FWLEVn) within the BDF file. See Understanding Document Assembly BDF Parameters in the Configuring a Document Assembly Application section of Designing for more information.

On initial display of the dialog, data item Levels 3 and higher within the data item hierarchy are expanded by default. Level 2 (the document level) remains collapsed. This view of the hierarchy enables the operator to quickly navigate through the documents and only view the first page of each document to verify that it is correct.

When the image displayed in the attachment viewer is not the first page of the document, the operator will know there is a problem with that document and take steps to correct it.

Navigation within the Data Items dialog is accomplished using keyboard keys and key combinations. Typically, pressing the CTRL key in combination with another key on the keyboard controls display of the Data Items dialog. This is the best method to access this dialog since many of the existing navigation functions on the Data Items dialog recognize CTRL key mapping conventions.

It is also possible to include the field data after each image name within the expanded document level. The INI parameter, DDL=# (Display Data Length) is used to include field data after the image name. When this parameter is set and the document level is expanded, the image line include the specified number of characters of data from the record as shown below:
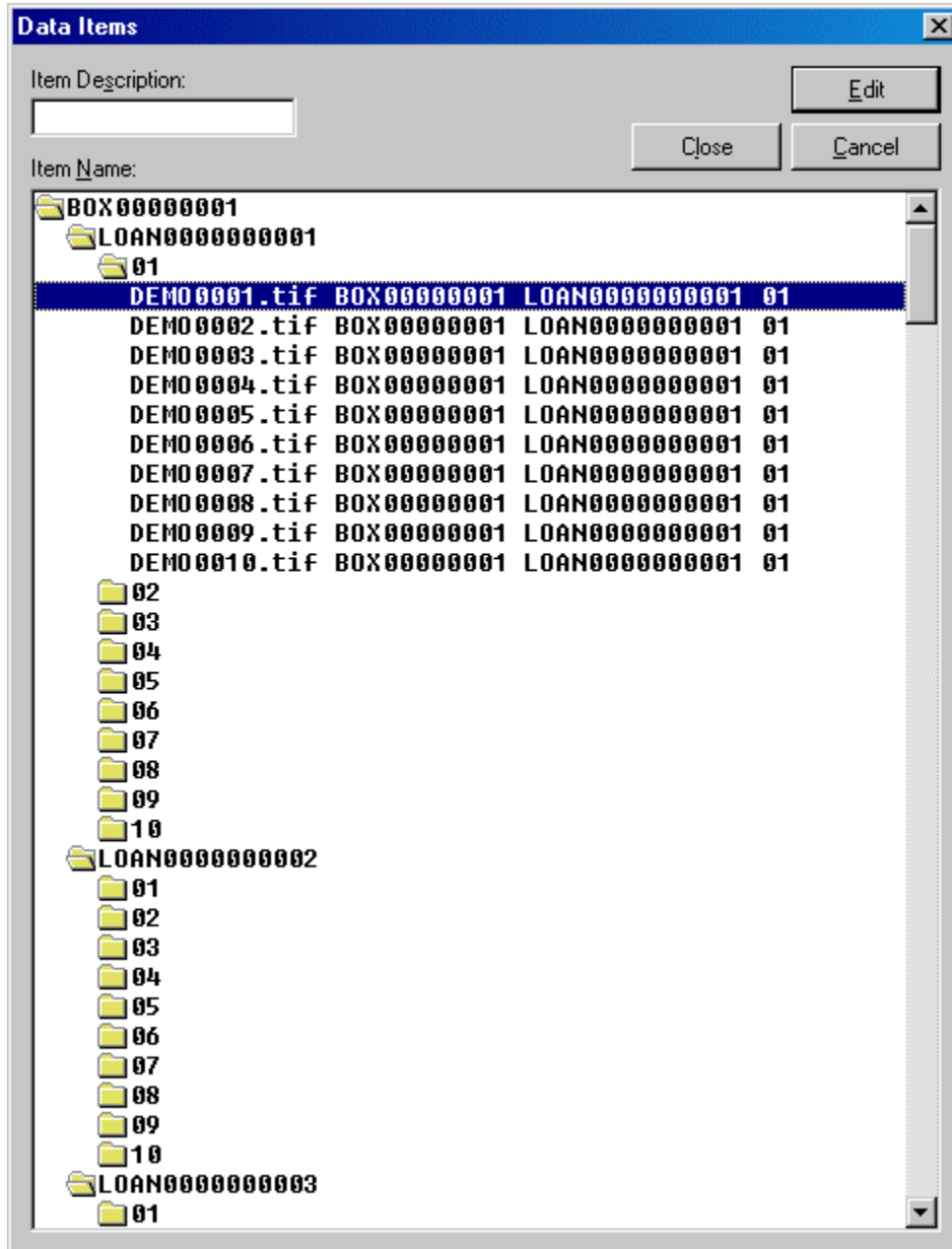
See Configuring a Document Assembly Application, Understanding Document Assembly Job INI Parameters in Designing for more information on the DDL= parameter.

To navigate the Data Items dialog:

1. In the **Item Description** field, type the name of a data item to highlight that item in the Item Name hierarchy.

2. Select the **Edit** button:
    o   At Level 2 or higher to collapse/expand the highlighted level folder within the Item Name hierarchy

     o   At Level 1 (page level) to close the Data Items dialog and position on the first field in the data form for that item.

3.   Select the **Cancel** button with the cursor positioned on any item within the Item Name hierarchy to close the Data Items dialog and give active control to the Completion window.

4.   Select the **Close** button to close the Completion session and exit (similar to the **File > Close** menu function on the Completion window).

5.   Navigate the Item Name hierarchy using the following keys on the keyboard:

    o   **Home** – Go to the first item in the hierarchy

    o   **End** – Go to the last item in the hierarchy

    o   **Down Arrow** – Go to the next item in the hierarchy

    o   **Up Arrow** – Go to the previous item in the hierarchy

    o   **Page Down** – Go to the next 'page' in the hierarchy

    o   **Page Up** – Go to the previous 'page in the hierarchy

    o   **Enter** – On a level folder to Open|Close that level in the hierarchy

    o   **Enter** – On an image name to display the form for that image and edit fields for the image.

Modifying Data Items in the Hierarchy

The purpose of Document Assembly is to verify the correct order of the batch and that the correct field data matches the current image. When either of these things is incorrect, the operator can reassemble the batch to reflect the correct order.

Note: The following functions do not have keyboard equivalents and may not be remapped.

To perform reassembly functions to the data item hierarchy, use the following keys:

- **Insert** – Insert an image and record
- **Delete** – Delete an image and record
- **Ctrl+Insert** – Insert an image only
- **Ctrl+Delete** – Delete an image only
- **Shift+Insert** – Insert a record only
- **Shift+Delete** – Delete a record only
- **Ctrl+I** – Insert a level only
- **Ctrl+D** – Delete a level only
- **Ctrl+X** – Cut an image and record
- **Ctrl+V** – Paste an image and record

Inserting and Deleting Images and Records

Consider the following as you insert and delete images and records in the hierarchy. Deleting an image physically deletes the image reference from the BDF as well as the actual image from its location on the hard disk. Remember that there are two distinct parts that make up the hierarchy being validated:

- The images referenced in the BDF and their level tags
- The field data in the batch records

When the batch is pre-built there is a one-to-one relationship between the images and the data records. However, when you insert or delete just an image (and not the data) it only impacts that part of the hierarchy. When you insert an image, the rest of the images move down in the batch, while the field data in the records remains the same. In the same manner, if you delete an image, the rest of the images move up in the batch, while the field data in the records remains the same. The same is true for records. It is possible to insert and delete both an image and a record at the same time, thus keeping the relationship between images and records the same as before, only including a new image and record or removing an image and record.

When you insert an image only (using Ctrl+Insert) or insert an image and a record (using Insert) the Image Insert dialog displays.
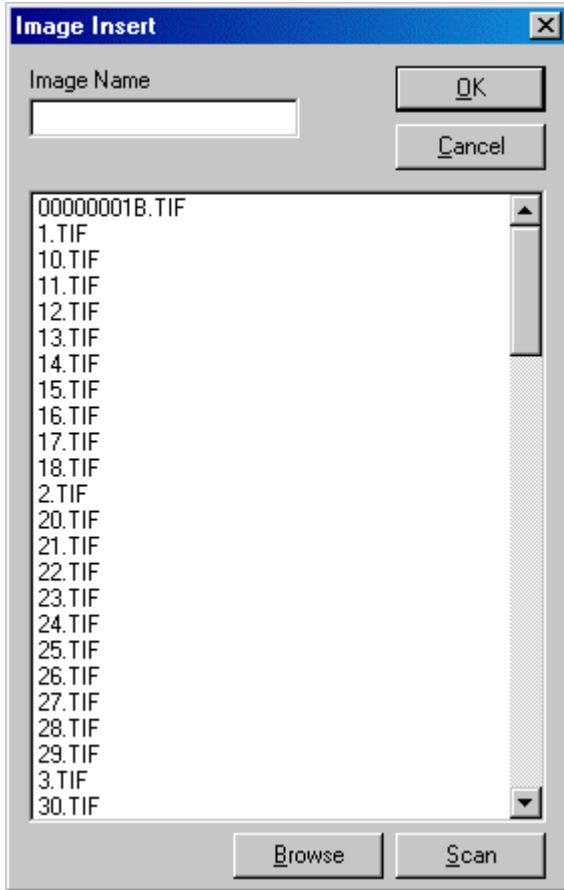
Perform the following on the Image Insert dialog:
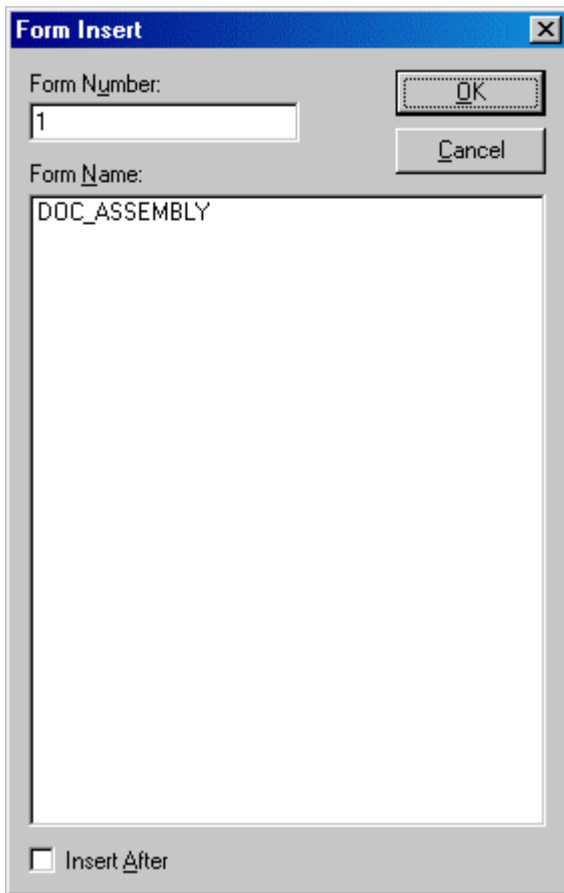
- In the **Image Name** field, type a valid image name and then select **OK** to insert this image at the highlighted position within the data item hierarchy.
- Select the **Browse** button to access the Open dialog where you can browse for an image outside of the default image path (typically the \Fwshare\FormWare\Images folder)
- Select the **Scan** button to access the Scan Operator program, where you can rescan an image using the selected image name.

Note: By default the **Browse** and **Scan** buttons on the Insert Image dialog are disabled. To enable these buttons, add the BOK=Y (Browse OK) and SCN=Y (Scan OK) INI settings to the System.ini or job INI file. See Configuring a Document Assembly Application, Understanding Document Assembly Job INI Parameters in Designing for more information on these parameters.

When you insert a record only (using Shift+Insert) or insert an image and a record (using Insert) the Form Insert dialog displays:

Perform the following on the Form Insert dialog:

- In the **Form Number** field, type the number of a valid form and then select **OK** to insert this form before the currently the highlighted position within the data item hierarchy.

- In the **Form Name** field, highlight a form name and select **OK** to insert this form before the currently highlighted position within the data item hierarchy.

- Check the **Insert After** checkbox to insert the selected form after the currently highlighted position within the data item hierarchy.

Note: When you select the option to insert both an image and a record, the Image Insert dialog displays first. Once you select an image from this dialog, the Insert Form dialog displays, allowing you to a form from the list of available forms.

Inserting and Deleting Levels in the Hierarchy

It is also possible to affect the level hierarchy of the batch by inserting and deleting levels. When you insert a new level into the hierarchy (higher than Level 1), the description of that new level is identified by an asterisk (*). For example:
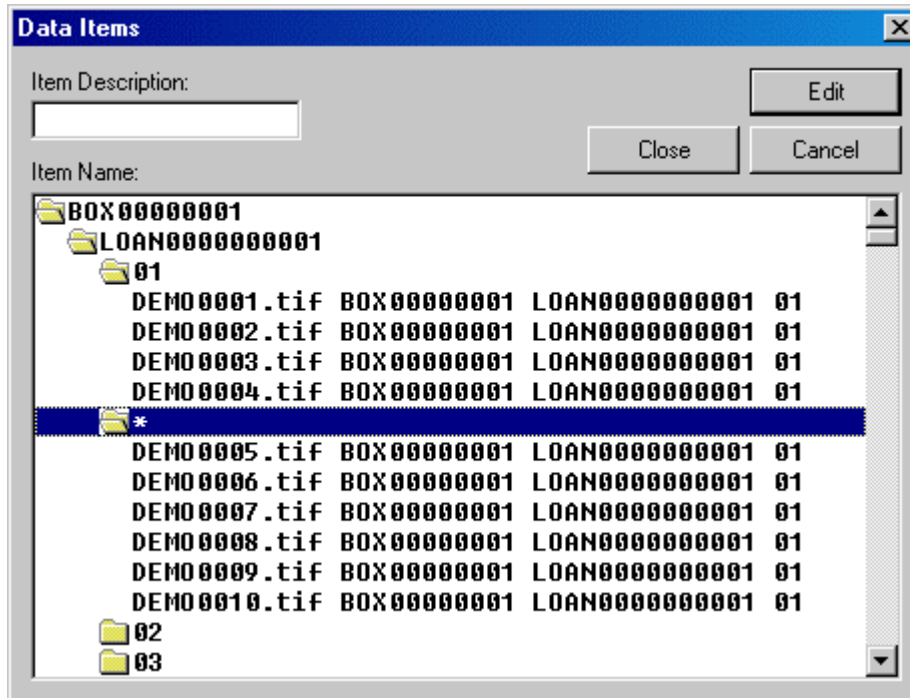
This behavior occurs within the FormWare Database because all level names within the Database must be unique. Within the FormWare File System, you can modify level names within the hierarchy using the VBA SetDataProperty method.

Using Distributed Processing

The FormWare distributed processing components provide the ability to share the workload among different physical locations not connected by a Local Area Network (LAN). Remote input clients and remote processing clients perform distributed processing in the FormWare system.

- **Remote input clients** create new work that enters a host site JobFlow for continued processing. The remote input clients, Distributed Scan, FormWare Filler, I-Forms, and XML Import, all push work into an existing JobFlow.

- **Remote processing clients** connect to an existing JobFlow at the host site. They receive transaction data from the host site, process it, and return the modified data to the host site for further processing.

Using the Remote Transfer Utility

You use the Remote Transfer Utility in the Distributed Scanning and Distributed XML processes. You can also call it as a stand-alone utility to perform a Remote Sync process or to transfer files with a BDF or XML extension.

You must perform the Remote Sync process to synchronize the application files on the remote client with the application files on the FormWare Server. The Remote Sync process synchronizes the user INI, group INI, and System.ini files. It also examines the user INI settings and updates all SCG files listed there.

The Remote Transfer Utility performs the following operations:

- **Remote Sync** – This process synchronizes application files on the remote client workstation with application files on the FormWare Server. The synchronization includes application files such as jobs and forms, as well as Scan Configuration files (SCG) and other FormWare-specific files required for distributed processing. The Remote Sync process examines the user or group INI files on the FormWare Server to determine which application files apply to a specific remote user. When a remote user logs in with a user name and password, Remote Sync uses that information to locate the corresponding user or group INI file. Remote Sync then attempts to synchronize the files referred to in the INI with those on the remote user's machine. Therefore, when you update an SCG, CFG, or JDF file on the server, all operators who use those files can obtain the updated version through the Remote Sync process. For more information on Remote Sync see the Configuring the File Server for Distributed Processing section in Installing & Configuring FormWare for more information.

- **Remote Scan** – Using this process, remote clients configured with scanners run the FormWare Scan Operator to create images and Batch Definition files (BDFs), which are then sent to a FormWare JobFlow via the Internet. See the Using Distributed Scan section for more information about scanning in a distributed environment.

- **Transfer File** – This process allows you to select specific BDF files or XML files to transfer from the remote client workstation to the server. When you run the Transfer Files process, the Open dialog displays. You can then locate the files you need to transfer, select them, and perform the transfer operation. The Open dialog restricts you to file types with a .bdf or .xml extension.

You can run the Remote Transfer Utility from the Windows Start menu or from the command line. By using a command line parameter, you can create desktop icons to perform certain tasks. You can use the following command line parameters with the Remote Transfer Utility:

- -h - Perform a remote sync operation. RMTXFER.exe –h

- -c - Perform a continuous file transfer. This option is equivalent to selecting the Remote Scan option on the interface. RMTXFER.exe –c

The Transfer Files option requires you to select certain files for transfer and therefore does not have a command line option.

Using Distributed Completion

It is possible to process work with Distributed Completion in on-line or off-line mode. Processing work in an on-line mode assumes that the remote client workstation maintains a constant dial-up connection to the Internet during processing. Once the operator completes the last batch of work, all work is automatically sent back to the host site through the open connection to the Internet.

Processing in an off-line mode assumes that after batches have been downloaded for processing, the remote Completion operator disconnects from the Internet while processing the work. Once the work is complete, a new connection to the Internet is automatically established and the work is sent back to the host site.

With the correct Internet connection settings, Distributed Completion can detect the absence of an Internet Connection and establish this connection prior to launching Distributed Completion. In addition, these settings allow the operator to key data in an off-line mode. When the work is complete, Distributed Completion looks for an active connection. If one is not found, then the operator is prompted to re-establish the connection to return the work to the FormWare Server.

Establishing a Dial-Up Internet Connection

Follow these steps to configure an Internet connection:

1. Launch Internet Explorer (IE).

2. From the Tools menu, select **Internet Options** to display the Internet Options dialog.

3. Select the **Connections** tab.

4. If you already have dial-up options, they display in the Dial-up settings area. If you do not have any dial-up options, select the **Add** button to create a new dial-up connection.

5. After adding a dial-up connection, highlight the name of the connection in the Dial-up settings area and select the **Set Default** button.

6. Select the **Always dial my default connection** option.

7. With the default connection selected, select the **Settings** button. IE displays the settings for your default connection.

8. In the Dial-up settings area of the Settings dialog, select the **Advanced** button to display the Advanced Dial-Up dialog.

9. Deselect the **Disconnect if idle for** check box.

10. Select the **Disconnect when connection may no longer be needed** check box.

11. Select **OK**. Continue to close the rest of the dialogs.

Running Distributed Completion

To run the Distributed Completion module:

1. From the Windows Start menu, select the **Programs > FormWare Remote > Distributed Completion** option. The Distributed Processing dialog displays.

2. Select **Remote Sync** to connect to the FormWare Server. The Remote Sync Connection dialog displays.

3. Type in the name of the **Web Server**, your **User Name** and **Password**.

Note: This step is only applicable the first time you use Distributed Completion. The RmtXfer.ini file populates with this information during the Remote Sync process eliminating the need to reenter this information.

---

4. Select **OK**. The synchronization process populates the Distributed Processing dialog with the JobFlows and JobFlow tasks that you have permissions to access.

5. Highlight a task that has Available Transactions and select **Process**. The Distributed Completion module displays with the task loaded for processing.

Note: A Completion license must be available at the host site in order for the workstation to receive work. Exiting Completion releases that license for another user.

Once you have finished keying your assigned transactions, all the work is returned to the host site along with an operator statistics file that is then appended to the enterprise-wide operator statistics information at the host site.

6. If you want to download another block of work, you can select the **Refresh** button on the Distributed Processing dialog to update the Available Transactions column and pick a task with available work.

Recovering from an Error

If a remote operator encounters an error, such as losing the Internet connection, Distributed Completion performs the necessary steps to recover the remote workstation. When the connection is re-established, the remote operator can then log into Distributed Completion and complete his work. If the remote operator encounters a problem that Distributed Completion cannot recover from, FormWare provides an Operator Recovery option. The syntax for this option is:

```
RMTPROC /r
```

Execute this option using the Start, Run command or create a desktop shortcut to perform the recovery automatically. Point the shortcut to the \FormWare\Programs folder on the remote workstation.

Caution: The Operator Recovery option should be used as the last step in recovering from an error. All work that has been keyed is lost when Operator Recovery is performed. The remote operator should inform the FormWare Administrator of the error so that any necessary changes, which need to be made to the FormWare Server or the FormWare application, can be addressed.

When you perform an Operator Recovery the following actions occur:

1. The program logs into the remote server.

2. The program searches the local workstation to look for a <UserName>.rwl file. If it finds the file, it removes all of the BDFs listed in the file, any subordinate files, and the RWL itself.

3. The program searches the FormWare Server to look for a <UserName>.rwl file. If it finds the file, it changes the status of each batch to Available and removes the RWL file.

4. The program removes the remote user's Completion license, freeing it for additional use.

For more information on recovering from errors in Distributed Completion see the Completion Tips and Tricks section of Troubleshooting FormWare.

Securing Information on Distributed Images

Add the TEI=Y INI parameter to the job INI file to control access to confidential data, such as Name and SSN, displayed on images sent to Distributed Completion operators. Enhanced image field clips are sent to remote Completion operators for processing instead of the entire image. Use Designer to create snippets of information for viewing. This parameter also deactivates the Full Screen viewing feature in Distributed Completion.

For more information on establishing security for Distributed Processing see the Establishing Security for FormWare section of Installing and Configuring FormWare.

Controlling File Transfers

When a remote user logs into Distributed Completion and selects batches in a JobFlow, by default all files that could be associated with each batch are copied to the client. In cases where the client does not need specific files and copying these files would cause significant overhead, you can exclude these files by adding the following parameters to the user INI file:

```
RMT_EXCLUDE_FILES=
{
    <WORKFLOWNAME>=
    {
            <TASKNUMBER>=XXXXXXX
    }
}
```

where <WORKFLOWNAME> is the name of the JobFlow and <TASKNUMBER> is the JobFlow task number you want to exclude from the download. Replace the XXXXXXX value for the task number with one or more of the following letters:

- D - Batch Data file
- I - Image file
- O - OCR file
- E - Enhanced Image file
- T - Thumbnail Image file
- C - Color Image file (JPEG)

For more information on INI file parameters specific to Distributed Processing, see the Configuring the File Server for Distributed Processing section in Installing and Configuring FormWare.

Using FormWare Filler

Many tools and commands in FormWare Filler are the same as in the FormWare Completion module. However, the FormWare Filler menu bar is a streamlined version of the Completion menu bar. On the File menu, the **Remote Sync** option is unique to the remote components. Other commands on the File menu enable the remote operator to enter data into a new file or append to an existing file.

FormWare Filler is designed for the casual user, rather than a traditional data entry operator. Therefore operator statistics are neither captured nor transferred back to the host site.

Note: New and Append are the only modes of Completion available with the FormWare Filler application.

To use FormWare Filler:

1. From the Windows Start menu, select Programs > FormWare Remote > FormWare Filler.

2. Select **Remote Sync** from the File menu. The Remote Sync File dialog displays.

3. Select the **Applications** check box and select **OK**. This launches the Remote Sync process, which downloads application files from the host site to the local machine. If you are performing a Remote Sync operation for the first time, the Remote Sync Connection dialog displays.

4. Type the **Web Server Name**, your **User Name**, and **Password**, and select **OK** to log into the system. After the connection with the Web Server is made, the Remote Sync dialog displays the progress. Once the Remote Sync process is complete, you are returned to the basic FormWare Filler screen.

5. To open work in FormWare Filler, select **New** from the File menu. The New File dialog displays.

6. In the **Job Name** field, type the job name or double-click on a configuration file in the **Job List** to display that name in this field.

7. In the **File Name** field, type the name of the batch file you are creating.

8. Select **OK** to display the form for the selected job.

9. Type data into the form. Additional records are automatically created when the current record is complete.

10. To close (terminate) a batch, select **Close** from the File menu or press [F3]-[A].

11. Select **Remote Sync** from the File menu. FormWare Filler displays the Remote Sync File dialog. The terminated batches that have not been sent to the host site display in the **File List** area on the Remote Sync File dialog.

12. Select one or more files from the File List to indicate you want them to be transferred to the host site and select **OK**. The Remote Sync process transfers the selected files. Once the transfer is complete, the files are deleted from the remote site, and you are returned to the FormWare Filler application.

13. Create a new file, append to an existing one, or exit FormWare Filler.

Using Distributed Scan

The following sections contain basic information on using the Distributed Scanning processes. These processes allow you to perform scanning tasks in a location that is remote from your FormWare Server and transmit images and batches to the server via the Internet.

The Distributed Scanning processes require specific components both at the remote client site and at the server site. The following sections describe the components installed at the remote location.

The Distributed Scanning components you install on the remote client workstation include:

- Scan Operator to scan in pages and create batches and image files.
- Scan Designer to modify scan configuration (SCG) files.
- License Manager (FWSLM) to provide access to the Scan Operator program.
- Remote Transfer Utility to transfer batches and image files to the FormWare Server.

You must also install the scanner hardware at the client site. Attach the scanner to the client workstation in the manner described in your scanner documentation and install scanner drivers.

Although you can use the Scan Designer program to modify existing scan configuration (SCG) files, be aware that the master copy of any SCG file resides at the host site. During a remote sync operation, the SCG files from the host site are copied to the remote site. Therefore, any changes made only at the remote site will be lost. Use the Scan Designer program only to make changes to the SCG files that are scanner specific. For example, you can set the contrast or intensity by modifying the SCG file at the remote site, testing the settings, and then sending the final SCG file to the host site for storage as the master copy.

Starting Scan Operator

To start the Scan Operator, select the Scan Operator option from the Windows Start menu (i.e., **Start > Programs > FormWare Remote > Scan Operator**). You can also create a shortcut on the desktop with the following command line:

```
\<ServerName>\FormWare\Programs\ScanOp.exe –A.
```

You must run the Scan Operator program with the –A command line parameter. This ensures that the name of the SCG used to create each BDF is written into the BDF as the first line. Once the BDF is transferred to the host site, the input client uses the name of the SCG to determine into which JobFlow step the BDF should be inserted. Without this information, the BDF and its associated images may get transferred back to the host site, but they will not be inserted into the JobFlow for continued processing.

> **Note**: The FormWare License Manager (FWSLM) must be active prior to launching the Scan Operator, and a blue hardware key (dongle) should be attached to the parallel port of the scanning station.

Transferring Files

The Distributed Scanning processes enable you to select files for transfer automatically or manually. To select files for automatic transfer:

1. From the Windows Start menu, select **Programs > FormWare Remote > Remote Transfer**. The Remote Transfer Utility dialog displays.

2. Select the **Remote Scan** option.

3. Select the **Perform Operation** button to begin the transfer.

If you perform the above steps before you begin scanning, the Remote Transfer Utility transfers each batch as it is scanned. The Remote Transfer Utility continually polls for completed batches between transfers. If you perform the above steps after you finish scanning, the Remote Transfer Utility transfers all batches it finds in the specified location when you select the **Perform Operation** button.

Perform the following steps to select files for transfer manually.

1. Scan images and batches using the Scan Operator.

2. Run the Remote Transfer Utility.

3. Select the **Transfer Files** option.

4. Select the **Perform Operation** button. The Open dialog displays.

5. Browse for the files you want to transfer.

6. Select the files you want to transfer by highlighting them.

7. Select the **Open** button to begin the transfer.

8. The Remote Transfer dialog closes automatically when the transfer completes.

When you use the Transfer Files process to transfer files to the server, the Remote Transfer Utility does not continue to poll for batches to transfer once the initial file transfer is complete.

After you transfer files to the server, the server sends a confirmation receipt back to the remote client confirming that it received the files. When the remote client receives the confirmation receipt, it deletes its copies of the transferred files. Thus, the remote scanning processes automatically perform clean up after the files transfer to the server.

Scanning Documents

The following section provides an example of the basic steps you take during Distributed Scanning to prepare files for processing. This example assumes that your job is already designed and that you have all necessary FormWare components installed.

1. Begin the Distributed Scanning process by running the Remote Transfer Utility.

2. Select the **Remote Sync** option and then the **Perform Operation** button to update system files on the client workstation with system files from the server.

3. Use your scanner and the Scan Operator to scan in batches of images. The Scan Configuration (SCG) file directs the Scan Operator where to store the new images and batches on the client workstation.

4. Run the Remote Transfer Utility again. Select the **Remote Scan** option and then **Perform Operation** to begin transferring files from the client workstation to the server. The Remote Transfer Utility looks for files in the location on the client workstation where the scanner put them. This location is specified by the `RMT_FILE_LOCATION=` parameter in the user's INI file.

5. The Remote Transfer Utility pushes the files to the Internet FormWare Server (IFS). The IFS receives the files and stores them on the FormWare Server in the location designated on the FormWare Internet Capture Configuration dialog (ICAPTURE.INI).

6. The server sends a confirmation to the remote client confirming that it received the files.

7. The remote client deletes its copies of the transferred files.

8. The transferred files are now on the server and ready for processing in a JobFlow.

Performing eEntry

You perform KFI and OCR character correction for eEntry through the IFS. The server directs work into a workflow and then routes it to remote clients. You view the work on a remote workstation via the Internet Explorer browser, key the data onto an I-Form, and return completed work to the FormWare File Server. eEntry caches images on the remote workstation and removes them when you return the associated batch to the server.

eEntry operates on the backbone of I-Forms, but differs from I-Forms in that it constitutes image-enabled KFI and OCR character correction. The following task sequence typifies a usual eEntry session.

- Access – The remote workstation accesses a workflow on the FormWare File Server.
- KFI – The remote operator keys data from the image onto the I-Form or corrects erroneously OCRed data.
- Send – After KFI occurs for all the images in the batch, you return the batch to the FormWare Server.
- Access – A new image stream (new batch) downloads automatically.
- Retry or Logout – If no batches are available for checkout, you see the "Retry or Logout" screen. Retry sends another access request, Logout disconnects the remote machine from the server.

Perform the following steps. The following steps assume the login sequence has been set up as two steps.

1. Launch Internet Explorer.

2. Select the appropriate URL or browse to the eEntry Login screen.

Note: Due to the nature of SSL, the URL you use to load the login/workflow selection pages must be https if the job posting URL is https.

3. Log in to an eEntry session by entering your user name and password for authentication. Your user name and password are case sensitive.

4. Select **Login** to display the Workflow Select web page. At the Workflow Select web page, select the workflow name and select an image queue (task number) to work on.

    The ALLOWJOBFLOW parameter in the user INI file and the workflow entered at the Workflow Select screen must be identical to the workflow name identified with the image queue on the server.

5. Select **OK** to open the image control.

6. Login occurs right before the first batch is checked out. Once you are authenticated at login, your request is directed to the web page that was specified at login. The session object on the FormWare Server streams the first image in the Completion queue to the remote client. The image control screen also displays the I-Form that was created to accompany the current workflow.

Note: FormWare denies access to eEntry if you are already logged in as a distributed user and returns a message stating, "User already has work checked out".

Because browser navigation is disabled by default, the image control screen is an IE view absent menus and toolbars. An eEntry task bar is provided and recommended for use during eEntry sessions. Refer to the Using Browser Navigation Tools section for information on enabling the IE task bar.

BDFs are checked out to one user at a time. You can configure your job to use a vertical or horizontal split between the image and the I-Form. The Image Control screen is split between the first image with toolbar navigation buttons and the I-Form, showing either OCR data for review and correction or blank fields for keying the data from the image.

Using Browser Navigation Tools

The Internet Explorer is the key-punch engine for eEntry sessions, not FormWare. Under normal eEntry operations only the IE title bar displays. All other IE toolbars are hidden. eEntry's control of the browser goes no further than field end edits, and it has no control of character-level options, navigation tools, or history features. Therefore, using the browser navigation tools during an eEntry session can cause problems. If you choose to use the browser navigation tools, the image control could become disabled. To avoid this problem, Captiva recommends using the eEntry image control toolbar options that are superimposed on the images that stream to you from the FormWare Server.

If necessary, you can enable the IE task bar by setting a system-level environment variable (TCKI_DEBUG=FE) or (TCKI_DEBUG=Y) in the System.ini file on the remote workstation and then restart the workstation. You may want to do this if you embed FormWare forms in existing HTML pages. This option cannot be set at the job level because once a session starts there is nothing that prevents you from keying different jobs. Therefore, if Job A had the buttons turned off and Job B had them turned on, you would have to destroy the browser session and restart, which eEntry does not allow you to do.

Using the Image Control Toolbar

Captiva recommends using the toolbar options on the image control screen and not the browser navigation tools. The image control toolbar buttons always display on the current image and are collectively, not individually, either enabled or disabled.
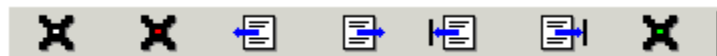
Following is a list of descriptions of the options on the image control toolbar, from left to right.

- **Interrupt and Logout -** Exit during a KFI batch and return the batch to available status with completed work saved. This is the same way Interrupt acts in Completion-New.

- **Cancel and Log out -** Cancel this session, send batch data to the FormWare Server, and log the operator out. This is similar to Interrupt and Logout, but no work is saved.

- **Previous Form -** Move back to previous image.
- **Next Form -** Move to next image.
- **First Form -** Move to first image in the batch.
- **Last Form -** Move to last image in the batch.
- **Close Current Batch -** Close batch and wait for more work without logging out.

If you configure your job to use a vertical split between the image and the I-Form, the toolbar may be only partly visible, depending on the screen resolution and the dynamics of the split. Also, the image control and the toolbar are active during a eEntry session, but eEntry disables the image control before any form is submitted and re-enables it when the next valid form or image loads.

Understanding the Image Control Context Menu

eEntry also supports limited manipulation of the image. Right click on the image to see the image context menu.Following is a list of descriptions of the options on the image control context menu, from top to bottom.

- **Zoom In -** Makes your view of the image 10% larger each time you select it, up to a maximum value of 500% larger.
- **Zoom Out -** Makes your view of the image 10% smaller each time you select it, up to a maximum value of 500% smaller.
- **Default Size -** Changes your view of the image back to 100% of its original size.
- **About -** Displays a product identifier and copyright window for eEntry Image Control.

Processing Transaction Data

When you log into an authorized job, the image control screen displays. You can now begin keying data from images or keying OCR edits as appropriate. Whether you are performing OCR character correction or KFI to blank fields, you work in the following eEntry modes.

- **New -** Starts a new batch for reject repair or data entry from image. A mode New queue could have many simultaneous operators working on the same task number. In KFI New mode you can interrupt a batch and exit by pressing the Interrupt button.
- **Append -** Opens an existing batch and enables new records to be processed at the end of the batch. Appending records from New mode is not supported. To append images you must run in Append mode.
- **Correct -** Starts a new batch for reject repair or data entry from image, and previously keyed data displays. A mode 'Correct' queue can have many operators working on the same task number simultaneously.
- **Read Only -** Opens the batch in Read Only mode and thus restricts keying. You can press either [Enter] or the [Down Arrow] key to move from field to field. In Read-Only mode you cannot change the data.

The mode of an eEntry session is determined at the server. Thus, the image control is unaware of what mode the session is in, which is why the toolbar always shows every button. Remote operators can either find out the mode from the FormWare administrator or can infer which mode is active by what they are allowed to KFI on the current image streams. Each mode performs a different part of the forms-processing solution, and the task number the administrator configures in the user INI file predetermines the working mode of the job. The ALLOWJOBFLOW parameter can thus authorize the same remote user to work in multiple modes.

Keying Data From Images

Images not processed through Recognition do not display OCR data in the I-Form fields you created. In that case, your assigned workflow is for KFI tasks, and you must key data from the entire image into the I-Form fields. Perform the following steps to perform KFI during an eEntry session.

1. Log into an eEntry session, and look at the current image name and number in the upper right corner of the Image Control screen. If another operator previously worked in the queue, the image number that displays is the image number where that previous operator left off.

2. Look at the first field clip highlighted on the image in the control screen and KFI the data you see highlighted in the corresponding blank field on the I-Form.

3. Tab to the next clip on the image and key the characters highlighted there into the corresponding blank field on the I-Form.

4. If you want to check your work, move back to a previous field by clicking the mouse to that field on the I-Form (browser dependent).

5. Continue tabbing through the field clips and keying the data you see. eEntry automatically advances to the next image when you tab out of the last field clip. As you advance to the next image, the image name and number on the toolbar updates.

6. When you finish working, select **Interrupt and Logout** to logout and return the image stream to available status or stay logged in by selecting **Close Current Batch** to close the batch and wait for more work.

Note: Logout occurs when you perform an eEntry logout (i.e., cancel transaction, interrupt, logout, browser close, etc.). As a security precaution, successful logout requires that you re-enter both your user name and password.

Keying OCR Edits

Images processed through Recognition, populates the fields in the I-Form with OCR data. By default, New and Append modes use the OCR data to populate the fields. Perform the following steps to key OCR edits during an eEntry session.

1. Look at the first field clip highlighted on the first image in the Image Control Screen. Assess the OCR results and make changes as needed. Be sure to examine every character on the I-Form against the image and key the correct characters over any erroneous OCR data.

Note: This OCR correction occurs independent of any field edits because no reject character flagging occurs in eEntry. Therefore, you must look at all the data for errors.

2. Tab to the next field clip and perform any necessary OCR correction. Tabbing through the clips occurs from top to bottom.

3. To check your work, move back to a previous field by clicking the mouse to that field on the I-Form (browser dependent).

4. Repeat the OCR correction until you have tabbed out of the current image and see the next image display in the control screen.

5. Repeat the above steps for each image until you complete the current batch (image stream).

6. When you finish working, select **Interrupt and Logout** to logout and return the image stream to "available" status or stay logged in by selecting **Close Current Batch** to close the batch and wait for more work.

6 Maintaining

Maintaining Overview

Once your FormWare File System is in production, system administrators must control the processing of work on a daily basis. The steps in this section guide administrators through the system maintenance tasks facilitated through the FormWare software. Maintenance tasks include:

- Understanding Log Files
- Monitoring a JobFlow
- Recovering a JobFlow
- Performing JobFlow Maintenance
- Performing JobFlow Task Maintenance
- Performing File Maintenance and Clean Up
- Reviewing Operator Statistics
- Generating QA Statistics Reports
- Creating or Updating a QA Statistics Database
- Administering Distributed Processing

Note: Captiva has implemented logging and diagnostic tools within the FormWare software to aid in supporting and testing the product. These logging files are for internal Captiva use only. Changes to or elimination of these files will be made without notification to customers.

Understanding Log Files

JobFlow Log Files

A log file is created with the JobFlow name and a .log extension. This is a fixed-field-length text file. Each time a batch is added, accessed, or removed from a queue, an entry is made in the JobFlow log. This file can increase in size rapidly; therefore, you need to create a backup copy and delete it on a regular basis. See the JobFlow Log File section for more information.

Operator Statistics Files

The purpose of the Operator Statistics file is to gather statistics that can then be used to create reports. The Operator Statistics file is generated automatically and includes data for operators on local clients as well as remote operators performing Distributed Completion tasks. Entries are added to this file each time a batch is interrupted or completed by an operator. To properly back up and maintain this information, you should add a step to your weekly process to run a report of the statistics and then rename the file with the current date. A new file will be generated automatically.

Batch Log Files

A batch log file is created when you run an FPL field or file edit program against a batch. You can identify batch log files by their .log extension. You can view the contents of a batch log file using a standard text editor. Double-click on the batch log file in an Administration module folder to launch the text editor and display the file contents.

Some companies use batch logs for error handling, writing only problems to the file. Others use batch logs to write reports of all accounts or provider numbers processed in a given batch. The creation of the batch log depends on the program that contains an output statement.

For more information on batch log files, see the Working with Batches section in Designing FormWare.

Monitoring a JobFlow

The JobFlow Monitor allows supervisory personnel to monitor one or more active jobs. It displays a visual "snapshot" of the active batches or transactions in the JobFlow. All pending, processing, and completed batches display for each task allowing supervisors to quickly identify processing bottlenecks, reallocate operators or machines to other tasks, and eliminate problems.

To monitor a JobFlow:

1. Open Administration and select the **JobFlow Monitor** button on the toolbar. The JobFlow Monitor window displays.

2. In the Job Browser, double-click on the name of the JobFlow you want to view.

   Each task displays as a line on the monitor window. The monitor line displays the percentage of batches completed and a bar that is divided into three sections. The section on the right (in red) indicates the batches pending or in need of processing. The middle bar (in yellow) indicates the number of active or in-process batches. The left side of the bar (in gray) indicates the batches that have completed the task. The monitor function updates every five seconds.

Recovering a JobFlow

JobFlow recovery is provided to help you recover JobFlow systems that have become corrupt. The JobFlow Recovery option that attempts to retrieve a JobFlow from an earlier version of FormWare or one that was interrupted due to a system or power failure. However, JobFlow Recovery may fail to recover a JobFlow if certain file corruption problems exist.

> **Note**: All JobFlow system users must exit the system before you start the JobFlow recovery process. While performing JobFlow recovery, you cannot access more than one JobFlow at a time.

To recover a JobFlow:

1. Open the Administration window and select the **Tools > Recovery > JobFlow Queues** option. The Task Recovery Utility dialog displays.

2. In the **Repair** area, select whether to repair **Complete JobFlows** or **Tasks** within the JobFlow during the recovery process.

3. In the **JobFlow Transaction Totals** area, select how you want to maintain JobFlow transaction totals during the recovery process. Options are:

   - **Do Not Change Totals** - Maintains all JobFlow transaction totals.
   - **Reset All to Current Total** - Purges all completed JobFlow transaction totals from the queue.

4. In the **Transaction State** area, select how to change the states of transactions in the JobFlow during recovery. Options are:

   - **IN PROCESS to AVAILABLE**
   - **FAILED to AVAILABLE**.

   This option also changes the state stored in the batch header, but does not affect the batch data. Use this option when there has been a power outage or loss of network connection during processing.

5. In the **Repair Logs** area, select whether to **Show Repair Logs** or **Print Repair Logs** at the conclusion of the recovery process.

6. In the **Show/Print Repair Logs** area, select whether to display or print repair logs **For All Repaired Tasks** or **For Repair Failures Only** (i.e., tasks that could not be repaired).

7. Select **OK**.

Performing JobFlow Task Maintenance

You can perform several functions on JobFlow task files using the options on the JobFlow task file context menu. You can distinguish JobFlow task files from JFD files by their numeric file name extension, which corresponds to the task number assigned during JobFlow creation. Access this context menu by highlighting any JobFlow task file within the Contents area of the JobFlow folder and right-clicking the file.

To maintain an entire JobFlow, see the Performing JobFlow Maintenance section.

Viewing JobFlow Properties for a Task

From Administration, you can view the properties of a specific JobFlow. To do this:

1. Open Administration.

2. In the File Browser, select the JobFlow folder.

3. Right-click on the file name of the task for which you want to view properties. Tasks are distinguished from actual JobFlow Definition files (JFD) by their numeric extension. The number indicates the position of this task within the JobFlow.

4. Select **Properties** from the context menu. The JobFlow Directory Properties dialog displays the file name, file path, size of the file, and the date on which it was last modified. for the task.

5. Close the window.

Modifying a Batch Property

You can modify a number of different batch properties from the JobFlow Maintenance dialog, using the menu items and toolbar options. Use the following instructions to change any of these properties for a batch:

1. Open Administration.

2. In the File Browser, select the JobFlow folder.

3. Right-click on a file name.

4. Select **Batch Maintenance** from the context menu. The JobFlow Maintenance dialog displays the batches assigned to that task.

5. Highlight one or more batches that you want to change.

6. Perform one of the following tasks:

    o In the **Transaction Priority** field, select a new priority for the batch and select the **Change Priority** button.

    o In the **Transaction Image Count** field, change the number to the correct number of images in the BDF and select the **Change Image Count** button. Because the image count is set when a batch is added to a JobFlow system, the number of images can change during production and no longer be in sync with the image count. An error message displays if the count does not match the actual count of the images in the BDF file.

    o In the **Transaction State** field, select a new state (e.g., Available or Hold) for the batch and select the **Change State** button.

    o Select **Delete** to remove the selected batch(es) from the JobFlow queue. This function does not delete the batch itself.

    o In the **Route Destination** field, select a new JobFlow task for the batch and select the **Route** button.

7. In the confirmation dialog, select **Yes** to perform your changes. The change will be reflected in the list of batches.

Maintaining Partial Batches

Launches the JobFlow Maintenance module and displays the selected batch within the JobFlow Maintenance Task dialog where you can perform partial batch maintenance functions on the selected task. To do this:

1. Open Administration.

2. In the File Browser, select the JobFlow folder.

3. Right-click on the file name of the task.

4. Select **Partial Batches** from the context menu. The JobFlow Maintenance dialog displays.

Repairing Tasks

You can repair a single task within a JobFlow rather than the whole JobFlow. The Task Repair function fixes the JobFlow queue, but does not reset the batch totals. To repair a task:

1. Open Administration.

2. In the File Browser, select the JobFlow folder.

3. Right-click on the file name of the task you want to repair.

4. Select **Repair Task** from the context menu. A process dialog displays and then disappears when the process is complete.

Modifying Task Properties

To modify the properties assigned to a task:

1. Open Administration.

2. In the File Browser, select the JobFlow folder.

3. Right-click on the file name of the task you want to view.

4. Select **Task Properties** from the context menu. The JobFlow Properties dialog displays for the task.

5. Make any modifications and select **OK**.

6. Select **Save** to save your settings.

Performing JobFlow Maintenance

The JobFlow Maintenance window allows you to view batches assigned to a JobFlow task, change batch priorities, re-route batches, and find batches based on various criteria. You access the JobFlow Maintenance dialog through Administration.

To modify an individual JobFlow task, see the Performing JobFlow Task Maintenance section.

Accessing the JobFlow Maintenance Window

To view the JobFlow Maintenance Window:

1. Open Administration.

2. In the File Browser, select the JobFlow folder.

3. If the JobFlow task files (files with a numeric extension) do not display in the Contents area for the JobFlow folder, select the **Show All Files** button on the toolbar to display all the files in the JobFlow folder.

4. Double-click on a JobFlow task file. The JobFlow Maintenance dialog displays the batches currently assigned to the JobFlow task, their state, priority, registration number, and number of images. The following columns display:

- o **Name** - Displays the full path of the BDF file currently assigned to the JobFlow task.
- o **State** - Displays one of the possible states of a batch in JobFlow. States include: Available, In Process, Processed, Failed and Hold. Hold indicates that the batch is held for this task and is not processed until the state is manually changed to Available.
- o **Priority** - Displays the priority number for the batch. Priority numbers are 1 to 1,000, with 1,000 being the first to be processed. You assign a priority number to a batch when the batch is assigned to a JobFlow task.
- o **Registration -** Displays the registration numbers used to group batches together for monitoring. You can assign up to eight registration numbers to a batch, with values of 1 to 1,000. Registration numbers are defined in JobFlow Designer.
- o **Image Count** - Displays the number of images listed in the BDF file. The image count is set when a batch is added to a JobFlow.
- o **User Name** - Displays the user name of the operator that last processed the batch.

The status bar displays the following information:

- o **Progress** - The left portion of the status bar displays the progress of any JobFlow Maintenance processes currently running. For example, a progress bar displays while a search is performed.
- o **Selected Transactions** - Displays the number of transactions selected in the JobFlow Maintenance window. You can reference this number after conducting a search to determine the number of transactions that meet the criteria.
- o **Total Transactions** - Displays the total number of transactions in the current JobFlow task.
- o **Total Images** - Displays the total number of images assigned to the current JobFlow task.

5. To view a subset of the list of batches, select from the following values and then select the **Select** button on the Transaction Selection toolbar. Transactions matching the search criteria are display in the batch list. If no transactions are found, the Selected Transactions option in the Status Bar displays zero and no batches display.
   - o **Transaction Name**
   - o **Priority**
   - o **Registrations**

6. To add or remove toolbars from the JobFlow Maintenance window, select or deselect any of the first five options on the View menu.

7. Select the **Refresh** option on the View menu to update the current list of batches in the JobFlow Maintenance window.

8. Select **Exit** from the File menu to close the JobFlow Maintenance window.

Searching for Batches

You can use the options on the Find menu on the JobFlow Maintenance window to search for batches in a JobFlow task, a JobFlow, or all JobFlows on the workstation. To search for a batch:

1. Open Administration.

2. In the File Browser, select the JobFlow folder.

3. Double-click on a file name. The JobFlow Maintenance dialog displays the batches assigned to that task.

4. From the Find menu, select one of the following options to display the Find Transactions dialog:

    o   **Transactions in Current Task**

    o   **Transactions in Current JobFlow**

    o   **Transactions in All JobFlows**

    The search criteria you set on the Find Transactions dialog are applied to the choice you made in the Find menu. The Find Transactions dialog displays the results of the last search conducted. This information is maintained per JobFlow Maintenance session.

Note: To select a range in the search fields, hold the [Shift] or [Ctrl] key while you select with your mouse.

---

5. To search by transaction name or pattern, type a search value in the **Transaction Name** field. Type the full path to the BDF you want to find or a pattern that represents a range of BDFs taking into account the pathname and extension of the BDF. You can use a plus (+) as a single wild card character or an asterisk (*) as an unlimited wild card character. You must type a value in the **Transaction Name** field before selecting **OK**.

6. To search for a specific registration number, select a value in the **Registrations** list.

7. To search by priority, select a priority value in the **Priorities** field.

8. To search by state, select a batch state in the **State** field.

9. When you have selected your search values, select **OK**. If you select more than one option in the Registration, Priority, or State criteria columns, an OR search is conducted for any of the numbers or states within the selection. If any criteria is specified for more than one option, an AND search is conducted, where at least one selection within each option must be found. For example, if you select registration numbers 3, 4, and 5, the transactions that have a registration number of 3, 4, or 5 are returned. If you then select the Available state as additional criterion, transactions that are available and have registration numbers of either 3, 4, or 5 are returned.

If you searched for **Transactions in the Current JobFlow** or **Transactions in All JobFlows**, the Results dialog displays the results of the search you conducted. Select a task in this dialog and select **OK**. The JobFlow Maintenance dialog displays the task you selected with the batches matching the search criteria highlighted.

Repairing a JobFlow

You select the Repair JobFlow task to perform recovery and repair on the selected JobFlow. You cannot be logged onto the JobFlow system during the repair. Performing repair from the menu does not reset the totals. The **Only Recovery** menu option resets the batch totals.

To repair a JobFlow:

1. Open Administration.

2. From the File Browser, select the JobFlow folder.

3. Right-click on the file name of the JobFlow for which you want to view the log.

4. From the context menu, select the **Repair JobFlow** option. The Task Recovery Utility dialog displays. See the Recovering a JobFlow section for information about this dialog.

Modifying JobFlow Properties

1. Open Administration.

2. From the File Browser, select the JobFlow folder.

3. Right-click on the file name of the JobFlow for which you want to view the log.

4. From the context menu, select the **JobFlow Properties** option. The JobFlow Properties dialog displays. The **Total Transactions** field displays the number of transactions currently in the queues for this JobFlow.

5. Select a registration number in the **Number** column to display the Registration Description dialog where you can type a description for the registration number. After typing a description for the registration number, select **OK** to return to the JobFlow Properties dialog.

6. In the JobFlow Monitor area, select one of the following options:

    o   **Independent Task View** - Uses the batches added to each queue rather than to all queues. If 3,000 total batches are added to the system, and the third task has 300 batches completed, 30 in process, and 170 pending, then the percentage calculations are based on 500 batches—not 3,000.

    o   **Historical Task View** - Uses the total of all batches added to the JobFlow system to calculate the percentage of batches completed. For example, if 3,000 batches are added to the first JobFlow task and the third task has only 300 batches completed, the percentage of batches completed is calculated using the 3,000 total batches. This is the default.

7. Select the **Log Transaction Activity** check box to define whether the JobFlow tasks are saved to the JobFlow log. If this option is not checked, no log file is generated for this JobFlow.

8. Select **OK** to return to Administration.

Maintaining the JobFlow Log File

A log file is created with the JobFlow name and a .log extension. This is a fixed-field-length text file. Each time a batch is added, accessed, or removed from a queue, an entry is made in the JobFlow log.

Viewing the JobFlow Log File

You can display the View JobFlow Activity Log dialog to review the transactions that have been processed by the selected JobFlow task. It contains the date and time the transaction status changed, the transaction name, its priority and registration numbers, the name of the task the batch was assigned to, the program which processed it (such as JFCLIENT or JFBATCH), and the path to the completed batch file.

To view the contents of the log file:

1. Open Administration.

2. From the File Browser, select the JobFlow folder.

3. Right-click on the JobFlow log file name.

4. From the context menu, select the **View** option. The View JobFlow Activity dialog displays.

5. Review the log information and select **Close**.

Viewing the JobFlow Log File Properties

To view the properties of the log file:

1. Open Administration.

2. From the File Browser, select the JobFlow folder.

3. Right-click on the JobFlow log file name.

4. From the context menu, select the **Properties** option. The JobFlow Directory Properties dialog displays the file name, file path, size of the file, and the date on which it was last modified.

5. Close the window.

Printing the JobFlow Log File

To print the contents of the log file:

1. Open Administration.

2. From the File Browser, select the JobFlow folder.

3. Right-click on the JobFlow log file name.

4. From the context menu, select the **Print** option. The Print dialog displays.

5. Select a printer and then select **OK**.

Deleting the JobFlow Log File

Every time a task moves through the JobFlow, four entries are added to the JobFlow Log file. Therefore, it grows large very fast. Captiva recommends that you set up a regular procedure for deleting this file; depending upon your volume you may want to delete the file daily or weekly. The file will recreate itself. You can turn off JobFlow logging; however, doing this reduces your troubleshooting ability if the JobFlow encounters errors.

Note: Before deleting the JobFlow Log file, back it up to a secure location.

Performing File Maintenance and Clean Up

Back up your FormWare environments on a nightly basis. It is important to have a well-tested backup plan. The system is dynamic; therefore, if your system fails you will lose everything from that point to the last backup. On your production system, backups should include, but not be limited to, your JobFlow and user and group INI files.

For optimal performance, you must periodically purge certain files from your system. If you allow the files within folders to accumulate continually, you can experience performance problems. If you must maintain a record of these files, you should archive them outside of the FormWare directory. We recommend that no folder ever contain more than 2,000 files. Allowing a folder to contain more than 2,000 files may significantly increase processing time. You should periodically purge files that are no longer used or needed from the following folders:

- **Files folder** – Batch Definition files (BDF) and error message files (ERR)
- **Images folder** – Image files (TIFF) and recognition results files (OCR)
- **Images.enh folder** – Enhanced image files (ENH)

- **JobFlow folder** – JobFlow Definition files (JFD) and JobFlow log files (LOG). One log file exists per JDF and bears the same name.
- **Stats folder** – Operator statistics log files (STA)
- **TMP folder** – All files in this folder. This folder is not accessible from the Administration module, but exists within the FormWare directory structure and can be accessed using Windows Explorer.

Note: You can also use the JobFlow Cleanup process to delete some, but not all, of these files on a regular basis.

Reviewing Operator Statistics

The Statistics folder contains operator statistics files. The Operator file within this folder contains statistics collected about operators during a Completion session. You can generate detailed reports from these files. When you select the Statistics folder to display its files, the Process menu displays the available options and reports you can run.

Reviewing the Operator Session Statistics

The Operator Session Statistics dialog displays information about an operator's performance during a Completion session for a batch. To view these statistics:

1. Run a batch through Completion.

2. From the View menu on the Completion window, select the **Statistics** option. The Operator Session Statistics dialog displays the following information:

The fields in the **Records** area, display the record statistics for the current operator and batch in the following fields:

   o **Entered** - Total number of records typed by the operator.
   o **Verified** - Total number of record correct sequences.
   o **Inserted** - Total number of records inserted into the batch with the Record Insert option.
   o **Deleted** - Total number of records deleted from the batch with the Record Delete option.

The fields in the **Errors** area, display the error statistics for the current operator and batch in the following fields:

   o **Character** - Total number of character corrections.
   o **Field** - Total number of field corrections.
   o **Record** - Total number of record corrections.

The fields in the **Time** area, display the time statistics for the current operator and batch in the following fields. Each time value is a portion of the total time spent within the Completion session.

   o **Keying** - Time spent in data entry mode.
   o **Help** - Time spent in help menu functions.
   o **Suspend** – Amount of time the operator suspended the statistics function with the [F3]-[S] keys.

The fields in the **Keys** area, display the keystroke statistics for the current operator and batch in the following fields:

- o **Gross** - Total number of all keystrokes for the operator during the Completion session. This total includes function keys.
- o **Net** - Total number of data entry keystrokes for the operator during the Completion session.
- o **Location** - Number of keystrokes typed in a field the first time that field was captured. This number does not include any characters rekeyed in the field.

The fields in the **Averages** area, display the averages statistics for the current operator and batch in the following fields:

- o **Gross** - Total number of keystrokes divided by the keying time.
- o **Net** - Total number of data entry keystrokes divided by the keying time.
- o **Location** - Total number of location keystrokes divided by the keying time.
- o **Record** - Records entered count divided by the keying time.

Processing Statistics Information

The Statistics Processing option on the Process menu displays the Statistics Processing dialog, which allows you to execute custom file edit or export programs against the statistics file. The resulting data are placed in the specified export device.

To run the statistics process:

1. Open Administration.

2. From the File Browser, open the Statistics folder.

3. From the Process menu, select the **Statistics Processing** option.

4. In the **Processing Types** area, select the type of program you want to process the statistics file: **Edit** or **Export**.

5. In the **Program Name** field in the Processing Options area, type the name of the program you want to perform statistics processing.

6. In the **Device Name** field, type the name of the device you want to perform statistics processing.

7. Select **OK**.

Converting Statistics

You can run a conversion program to convert dates and long file names. By default, the date format for statistics files is six digits, containing a two-digit year. To avoid problems when the six-digit format reaches or exceeds the year 2000, you can choose either a six-digit (011299) or an eight-digit (01121999) format. To use the eight-digit format for statistic files, make sure your System.ini file includes the YEAR=4DIGIT parameter. Statistics files created once the parameter is changed will then permit a change to an eight-digit date format.

By default, all filenames (including job names) in FormWare are limited to eight characters with a three-character extension (8.3). You can extend this limit by adding the LFN=Y parameter to the System.ini file.

> **Caution**: The options on the Statistics Conversion Program window do not convert existing statistics files. They only prepare the FormWare system to create statistics files as specified from this point forward. You should back up any existing statistics files into a separate directory before you set any options on this window to avoid potential file corruption problems.

To run the conversion program:

1. Open Administration.

2. From the File Browser, open the Statistics folder.

3. From the Process menu, select the **Conversion Programs – Dates and Long Files Support** option. The Statistics Conversion program window displays.

4. In the entry field, type the number that represents how you want to convert the date and file name formats for all new statistics files. Options are:

    o **1. Convert the statistics to an 8-digit date and long file/job names** - Converts all dates in statistics files to an eight-digit format (MMDDYYYY) and permits the use of long names (up to 27 characters) for all files and jobs in the FormWare system. To use this option, you must set the YEAR=4DIGIT and LFN=Y parameters in the System.ini file.

    o **2. Convert the statistics to a 6-digit date and long file/job names** - Converts all dates in statistics files to a six-digit format (MMDDYY) and permits the use of long names (up to 27 characters) for all files and jobs in the FormWare system. To use this option, you must set the LFN=Y parameter in the System.ini file.

    o **3. Convert the statistics to an 8-digit date and short file/job names** - Converts all dates in statistics files to an eight-digit format (MMDDYYYY) and restricts the file and job names in the FormWare system to 8.3 characters. To use this option, you must set the YEAR=4DIGIT parameter in the System.ini file.

    o **4. Convert the statistics to a 6-digit date and short file/job names** - Converts all dates in statistics files to a six-digit format (MMDDYYYY) and restricts the file and job names in the FormWare system to 8.3 characters.

5. In the **Continue** field, type a Y to accept the conversion option you selected and proceed with the conversion.

6. Select **OK** through the processing dialogs until the process is complete.

Reviewing Operator Statistics Reports

You can generate the following reports based on operator statistics:

*   **Operator Batch Detail Report** - Provides general statistics for each batch record, including batch name, mode of Completion, Completion session start and end times, operator key and non-key time, gross and net keystrokes, etc.

*   **Operator Job Summary Report** - Provides general statistical information totaled by Completion mode within each subdivision, including batch creation date, mode of Completion, operator key and non key times, gross and net keystrokes, gross and net keystrokes per hour (KSPH), etc.

*   **Operator Keystroke Report** - Provides gross and net keystroke and record count information and averages totaled for each subdivision of the records, including batch name, mode of Completion, keying time, record count, records per hour, gross and net keystrokes, gross and net keystrokes per hour (KSPH), possible characters, location characters, etc.

- **Operator Summary Report** - Provides time, record, keystroke, and averages totaled by subdivision, including batch creation date, total time, keying time, record count, gross and net keystrokes, error count, gross and net keystrokes per hour (KSPH), records per hour, etc.
- **Operator Time Report** - Provides operator time statistics for each record in the selected parameters including batch name, mode of Completion, batch start and end times, total batch time, keying time, help time, suspend time, record count, etc.

To generate a report:

1. Open Administration.

2. From the File Browser, open the Statistics folder.

3. From the Process menu, select the option for the statistics report you want to run. The Operator Statistics Report window displays.

4. Type the criteria for the report you want to generate.

5. In the **Is the entered information correct** field, type a Y to generate the report.

Reviewing the Statistics File

The statistics file is a FormWare batch. This means that you can use the FormWare programming language (FPL) to create customized statistical reports for your applications. The programs for the standard reports are in the \FormWare\Export folder and can be modified for your specific needs.

The operator statistics file is information compiled in the job OPRSTATS each time an operator suspends, interrupts, or terminates a file. The following table details the fields and field lengths of the data elements in the OprStat1.fdf:

Data Element	Field Length	Data Element	Field Length
Operator Name	10	Batch Status	1
Batch Name	10 or 27	TTY Name	10
Completion Mode	1	Reserved field	14
Batch Creation Date	6 or 8	Number of records corrected	4
Keying Time	6	Number of records rejected	4
Gross Keystrokes	8	Total number of corrections	5
Net Keystrokes	8	Reserved Field	4
Number of records	5	Total number of field characters	8
Number of record corrects	5	Total number of location characters	8
Number of field corrects	5	Time operator suspended batch	6
Number of character corrects	5	Total time	6
Job Name	10 or 27	Total amount of non-keying time	6
Number of inserted records	5	Average gross keystrokes per hour	5
Number of deleted records	5	Average net keystrokes per hour	5
Number of characters deleted	5	Average location keystrokes per hour	5

Data Element	Field Length	Data Element	Field Length
Number of reset errors	8	Average total keystrokes per hour	5
Time operator started batch	6	Average records per hour	4
Time operator closed or interrupted batch	6	Number of documents completed	4
Time operator displayed help functions	6	Reserved field	16

Using Operator Statistics Data with VBA

The following table lists the available operator statistics data, corresponding VBA function, and description. See the VB for FormWare Reference Manual for more information.

Operator Statistics Field Label	VBA Function	Description
Operator Name	StatsOperator	Completion operator logon name
Datafile Name	StatsFile	Completion data file name
Mode	StatsMode	Completion mode
Date	StatsDate	Date of collected statistics. Keying over midnight will have the next day's date.
Keying Time	StatsKeyTime	Time operator able to key while in session. Non-key time is time displaying menu functions such as searching or viewing OCR data. Suspend time is time that operator pauses with [F3]-[S]. Keying time is total time minus key time. The format stored is hhmmss, so 1 minute 59 seconds = 000159
Gross KS	StatsGrossKeys	The total keystrokes pressed during session. Pressing a key combination such as [F3]-[R] counts as one function, equaling one gross keystroke. Mouse clicks on Completion menu will count as one gross keystroke.
Net KS	StatsNetKeys	Net keystrokes are the actual characters captured in a field. Keystrokes such as Enter, arrows, Page Up, etc only count as gross keystrokes.
Number of Records	StatsRecords	Records captured
Record Corrects	StatsRecordsCorrected	The number of [F3]-[F9] functions pressed in Correct or Verify mode.
Field Corrects	StatsFieldsCorrected	The number of [F4]-[F9] functions pressed in Correct or Verify mode.
Character Corrects	StatsCharactersCorrected	The number of [F9] functions pressed in Correct or Verify mode.
Job Name	StatsJob	The job name for the batch.
Inserted Records	StatsRecordsInserted	The number of [F3]-[Ins] functions used to insert records into an existing batch.
Deleted Records	StatsRecordsDeleted	The number of [F3]-[DEL] functions used to delete records in an existing batch.
Characters Deleted	StatsCharactersDeleted	The number of [DEL] functions pressed.
Reset Errors	StatsKeyErrors	Not valuable information in Windows program – in file from legacy systems.

Operator Statistics Field Label	VBA Function	Description
Start Time	StatsStartTime	The time that the statistics gathering session was initiated.
Finish Time	StatsStopTime	The time that the statistics data was written.
Help Functions Time	StatsHelpTime	Time displaying menu functions such as searching or viewing OCR data. This time plus suspended time equals non-key time.
Status		The batch was interrupted or terminated.
TTY Name	StatsTTY	Not needed for Windows systems – in file from legacy systems.
Unused		Available to write to operator file using FPL @STOINSTA
Records Corrected	StatsRecordsWithCorrects	The number of records that had one [F9] keystroke sequence used during Correct or Verify mode. No matter the [F9] sequence or the number, this number will increment just once per record.
Records Rejected	StatsRecordsRejected	The number of [F3]-[R] functions used to reject records. These will not increment if you use custom reject record code.
Total Corr Sequences	StatsTotalCorrects	The total of any type of [F9] functions. This equals record corrects plus field corrects plus corrects.
Reserved Field		
Total Fld Characters	StatsTotalCharacters	The total possible characters that could be captured during the session. If each field is 10 bytes and there are 4 fields on the form, then the total will be 40 ks per record.
LOC Characters	StatsLocationKeys	The Net Keystrokes will be counted just the first time the field is captured. If an operator re-keys field data, the net keystrokes will increment, but the LOC (location) keystrokes will not.
Suspended Time	StatsSuspendedTime	The total time that the operator has engaged the suspend function with [F3]-[S].
Total Time	StatsTotalTime	The total time, stop time minus start time. This will also be keying time plus nonkey time.
Nonkeying Time	StatsNonKeyTime	This is the total suspended time and help time.
Avg Gross KSPH	StatsAverageGross	In this session, the KSPH calculation for gross keystrokes.
Avg Net KSPH	StatsAverageNet	In this session, the KSPH calculation for net keystrokes.
Avg LOC KSPH	StatsAverageLocation	In this session, the KSPH calculation for LOC keystrokes.
Avg Total KSPH	StatsAverageCharacters	In this session, the KSPH calculation for total possible keystrokes.
AVG Recs Per Hour	StatsAverageRecords	In this session, the KSPH calculation for records.
Document Count	StatsDocumentCount	If using document indicator, will count instead of records in the operator statistics file. In VBA, both numbers are stored.
Unused		
	StatsKeyErrorsCorrected	Not used.

Generating QA Statistics Reports

The QA Statistics database allows you to capture, store, and generate reports on detailed QA statistical data. This includes the actual raw data that is collected by OCR. By analyzing this information, administrators can identify and distinguish OCR engine errors from operator data entry errors. This data also identifies in what process the error occurred. The raw is captured before and after each task allowing side-by-side comparisons of data as it moves from one task to the next. Due to the flexibility of FormWare and its compliancy with SQL, you can query the QA Statistics database and create your own reports using a third-party reporting tool (e.g., Crystal Reports).

Generating a QA Statistic Detail Report

Using the QA Statistics Reporting function, you can generate a QA Statistic Detail Report from the data stored in the QA Statistics database.

The following statistics are captured within the QA Statistic Detail report:

- Date/Time
- Operator name/ID
- Image file name
- Data item name
- Field name
- Process/Task name (OCR, Post-Completion Edit, Completion Mode, etc.)
- Modified field contents at the end of each process

To generate this report:

1. From the desktop, select Start > Programs > FormWare > QA Statistics Console.

2. Select the **Reporting** drawer on the program bar.

3. Select the **Reports** icon to display the Reports window.

4. Select the + sign by QA Statistics Reports in the tree view to display the available reports.

5. Highlight the **QA Statistic Detail** node and right-click on it to display the context menu.

6. Select the **Generate Report** option to display the Generate Report dialog for this report.

7. In the **Report** title field, type the title of the report that will display at the top of the report when it is generated. This enables you to uniquely identify the name of each report you generate. For example, the title of the QA Detail Report can be "QA Detail Report for East Coast Facilities"

8. In the **Sort** by area, select the option that determines how data is organized on the report. Options are:

    *   **Field Name** – Report data is organized alphabetically by field names on the form. This is the default option.

    *   **Keying Order** – Report data is organized numerically by field keying order on the form.

    *   **Field Type** – Report data is organized alphabetically by field type.

9. In the **Date** area, select the radio button next to the **From** field, to select a specific range for the report. In the **From** and **To** fields, select the values that determine the range of time for data on the report. The date format that displays in this field (either MM/DD/YY or MM/DD/YYYY) is determined by the format set on the workstation's control panel.

    In the date fields, select the drop-down arrows to display a month-to-month calendar where you can select a starting and ending day for the range to include in the report. The **From** date fields default to January 1, 2001 so that you can include all data in the database. The **To** date defaults to the current date.

    Or, select the radio button next to the **Last** field, to select a number of days for the report. In the **Last** field, type the number of days from the current day to include in the report. The default is 7 days. The maximum value is 999.

10. In the **Select forms** area, select the specific forms to include in the report or select the **All forms** checkbox to include all forms that have been enabled for QA statistics in the report.

11. In the **Select tasks** area, select the specific tasks to include in the report or select the **All tasks** checkbox to include all tasks that have been enabled for QA statistics in the report.

Note: Selecting the All tasks option may require too much processing overhead for the report generator and an error message will display. Try to make the report criteria as specific as possible to prevent this from occurring.

---

12. In the **Select operators** area, select the specific operators to include in the report or select the **All operators** checkbox to include all operators that have QA statistics in the report.

13. In the **Enter data items** area, select the **All data items** checkbox to include all data items that have QA statistics in the report or select **Find** to display the Search Data Item dialog. See Searching for Data Items for more information.

14. Select the **Preview** button to generate the report and display the results for visual review. See Previewing Reports for more information.

15. Select the **Print** button to display the Print dialog, where you can send the report to an available printer.

16. Select the **Save As** button to display the Save As dialog, where you can browse for a folder location to save this report as a TXT file.

17. Select the **Cancel** button to cancel your changes on the Generate Report dialog and return to the main Reporting window.

Previewing QA Statistics Reports

Once you generate a QA Statistics report, you can preview the report results on your screen. To preview the report:

1. Select the **Generate Report** option to display the Generate Report dialog.

2. Select the options for your report.

3. Select the **Preview** button. The Preview dialog displays.

4. Use the following toolbar buttons on this dialog:

   - **Close Current View** – Close the current drill down view. This button is active only when drill down view is active.

   - **Print Report** – Display the Print dialog where you can select a printer and generate a printed copy of the report.

   - **Export Report** – Save the report as a text file with the name you specify.

   - **Refresh** – Update the report with current data according to the settings you made on the Generate Report dialog.

   - **Toggle Group Tree** – Displays the list of images within the current batch. Select an image in the tree to display statistics detail for that image.

   - **Zoom** – Change the size of the report detail display within the Preview dialog.

   - **Go to Page** – Advance or move backwards through the pages in the report. Options are Go to First Page, Go to Previous Page, Go to Next Page, and Go to Last Page.

   - **Search Text** – Display the Search dialog to look for a specific item on the report.

Note: All toolbar buttons have a corresponding option on the context menu. Right-click to display the context menu and then select your option.

Here is an illustration of the Preview dialog with the Toggle Group Tree button activated:
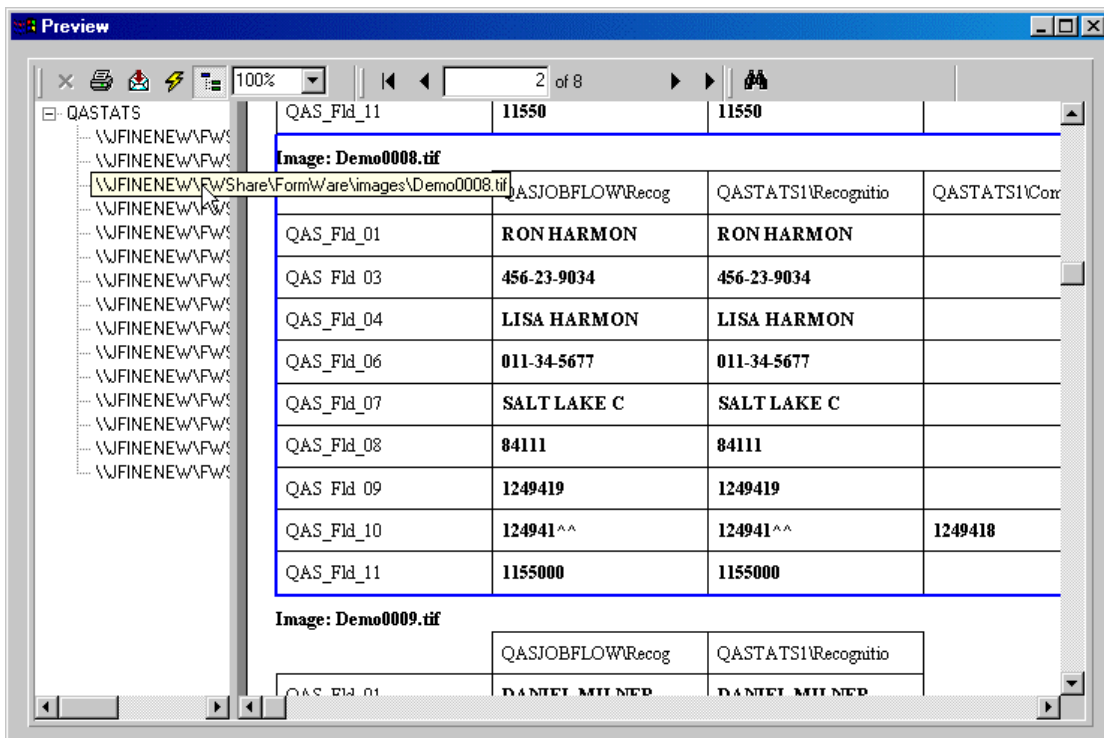
Searching for Data Items

You can limit the data on the QA Statistic Detail Report by specifying the data items you want to appear on the report. To search for specific data items:

1. From the Generate Report dialog, select the Find button in the Enter data items area to display the Search Data Item dialog. Use this dialog to specify the search criteria.

2. In the **Search what** field, type any alphanumerical values to limit the data item search to include only those data items containing these values. This field works with the **Criteria** field setting to determine if the data items returned must Begins with, Ends with, or Contains the character or character set within this field.

Note: Wildcard characters (@, *, etc.) are not valid entries for this field.

3. In the **Max** field, type the maximum number of data item results to return for this query. The maximum is 2000.

4. In the **Criteria** field, select the option that determines whether the data items retrieved in the search should **Begins with**, **Ends with**, or **Contains** the character(s) in the **Search what** field.

5. Select the **Start Search** button to run the query. Search results display in the **Results** section of this dialog. The number of data items located based on the search criteria display in the **Found** area. The total number of data items found displays in the **Total found** field.

6. Highlight one or more values in the **Found** list and select **Add** to add them to the **Selected** list.

7. Select **OK** to add the selected items to the list of data items on the Generate Report dialog.

Purging the QA Statistics Database

Collection of statistical data on OCR results and operator data entry can create huge amounts of data in a very short time. Therefore, periodic purging of the data within the QA Statistics database is crucial to proper maintenance of the volume of data that collects there.

To purge data from the QA Statistics database:

1. From the desktop, select Start > Programs > FormWare > QA Statistics Console.

2. Select the **QA Purge** icon to display the QA Stats Purge dialog over the QA Statistics Console.
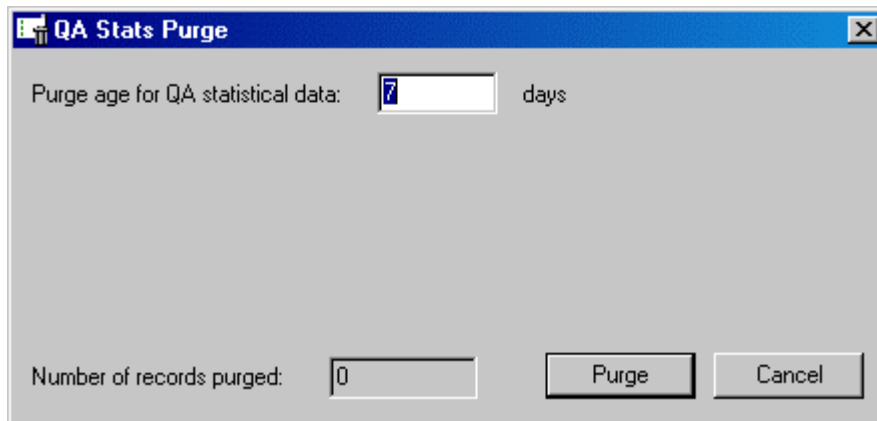
3. In the **Purge age for QA statistical data** field, type the number of days old that data must be purged from the database. Any data older than this numbers of days is purged.

4. Select **Purge** to start the purge process. Status information about the purge process displays on the dialog.

5. When the purge process is complete, the **Number of records purged** field displays the number of records removed from the database and a confirmation dialog displays.

6. Select **OK** on the confirmation dialog and return to the QA Statistics Console.

Administering Distributed Processing

Recovering From a Distributed Completion Error

Remote operators can experience problems when a Distributed Completion session fails and cannot recover on its own. In this situation, you can use the Operator Recovery option. The syntax for this option is:

```
RMTPROC /r
```

You can create a desktop shortcut for this option that points to the \FormWare\Programs directory on the remote workstation.

For more information on recovering from errors within Distributed Completion, see the Completion Tips and Tricks section of Troubleshooting FormWare.

7 *Troubleshooting*

Troubleshooting Overview

As with any software implementation, you may have questions or need help while developing your FormWare application and using it in a production environment. This section provides some general technical tips as well as steps for contacting Captiva Technical Services. See the following topics for more information:

- Training and Customer Support
- Getting Help
- Scanner Tips and Tricks
- Registration Tips and Tricks
- Recognition Tips and Tricks
- Completion Tips and Tricks
- System Administration Tips and Tricks

Training and Customer Support

Captiva Software Corporation is committed to having a satisfied customer base. This commitment begins with a training program tailored to address each user's specific needs and facilitate the implementation effort. There are several courses from which you can choose depending upon your requirements. Please visit our web site at www.captivasoftware.com or call the Inside Sales department at (858) 320-1100 for course descriptions, training schedules, and course registration.

Training is supplemented by Captiva Software's Technical Services program. This program provides extensive continuing assistance in the form of telephone support, new releases, software updates, and access to our eSupport Self-Help Center.

For additional information on Technical Services, please contact your local Sales Representative.

Getting Help

If you are a supported customer and you experience any problems that require assistance, you can contact Captiva Software Technical Services using one of the following methods:

- Voice: (435) 655-7350
- Fax: (435) 645-9610
- E-mail: fwsupport@captivasoftware.com

For International customers, please contact your Captiva Regional Office for answers to problems or questions.

Before Calling Technical Support

If you encounter a problem while using a Captiva product, please review the questions below to isolate the cause. Sometimes this analysis could lead you to a solution.

- Was the function working before?
    - What has changed since it last worked?
    - Can you reinstate the correct working environment?
    - What are the expected results?
- Can the problem reliably be recreated?
    - Which files are needed to recreate the problem?
    - Will a simple application show the same problem?
- Does the problem still exist when you remove:
    - Rules?
    - VBA Projects?
    - Field or File Edits?
    - Keyboard Maps?
    - INI Settings?
- Is the problem related to a specific batch or image?
    - What is unique about this batch or image?
- Is the problem with a specific form?
    - How is the form unique?
    - Can other forms be changed so they will exhibit the same problem?
- Is the problem related to a specific job?
    - How is the job unique?
    - Can working jobs be changed so they will exhibit the problem?
- Does the problem occur when testing from:
    - Designer?
    - Job level?
    - Form level?
    - Field level?
    - Administration?

   o JobFlow?

   o Using the .ocx?

   o Using the .exe?

Submitting an Issue to Technical Support

When you call, you should be at your computer and be prepared with the following information:

- The version number of ClaimPack you are currently using.

- The version number of FormWare you are currently using.

- The type of hardware being used, including operating system and network.

- A full description of the software's behavior, including a description of what you were doing at the time the problem occurred. Please remember to note the modules you were working with, and any other information needed to reproduce the problem.

- The exact wording of any messages that were on the computer screen.

- A full description of how you tried to address the problem.

If you need to submit a customer problem to Technical Services for analysis, you must compile and prepare files pertaining to this problem in the following manner:

1. Create a zip file with the minimum number of files necessary to duplicate.

2. Make sure to adjust for differences in devices, programs, etc., so that the problem can easily be seen when we locate the files in a different drive. For example, instead of hard coding a path as G:\FORMWARE when your environment variable and path statements are set to G:\FORMWARE, use %FORMWARE%. This eliminates the need to change the path for every program and device when a technical support representative at Captiva Software is attempting to duplicate the problem.

3. Zip these files with their folder names with a path to one level below the FormWare directory. For example, \files\1040EZ.bdf. This will allow us to drop it into a C: drive or onto our network, if needed. Please do not use self-extracting zip files.

4. Provide a clear set of step-by-step instructions for duplicating the problem. We are not as familiar with the job as you are, so your assumptions will not necessarily be obvious to us.

Depending on what the problem is and where it appears, you may need to submit the following files:

Folder or Files	Description
Devices	Device (.dev) associated with the job on the Job Properties, Export tab.
FileEdit	File Edit associated with the job on the Job Properties, Edits tab.
Files	<filename>.bdf and the <filename> The <filename>.bdf contains the paths to the images used when the data file is created in Completion. The <filename> with no extension is the data file created during Completion.
FldEdit	Field Edit associated with the job on the Job Properties, Edits tab.
Forms	<formname>.fdf and <formname>.tif
Images	<imagename>.tif in the BDF
INI	System, Job, User, and others, if needed

Folder or Files	Description
JobFlow	<jobflowname>.jfd – JobFlow Definition File <jobflowname>.001 task 1 <jobflowname>.002 task 2 <jobflowname>.003 task 3, etc. <jobflowname>.log – JobFlow Log file
Jobs	<jobname>.jdf
Output/Export	The export program for the job on the Job Properties, Export tab.
VBA	<vbaproject>.vbp

Scanner Tips and Tricks

Maximizing Scanner Throughput

This section contains a list of helpful troubleshooting items to enable your scanner to function at its fully rated speed. This is not a complete list of hints. If you try these suggestions and are still experiencing problems with the performance of your scanner, please contact the scanner manufacturer.

- Most scanner performance ratings are for 8½ x 11 documents creating 200 DPI images. Using a higher resolution or a larger paper size will cause the scanner performance to degrade. European customers who use the A4 paper size should be aware that this size of document takes approximately 10% longer to scan.

- For high-volume scanners with automatic document feeders (ADF), the rollers for these feeders must be cleaned regularly. You may need to schedule scanner cleaning on a weekly or monthly basis depending on the volume of documents you are scanning.

- A scanner's bulb can become weaker with age. Some scanners will automatically recalibrate themselves to compensate for this circumstance but it is not always successful. A weak bulb can produce a poor-quality image.

- The glass platen must be cleaned periodically. Dirt on the platen can result in noise on the image. A noisy image requires more disk space and therefore, takes longer to write to the disk.

- The scanner firmware must be current. If you are using old firmware, many of the features of the scanning software may not be supported.

- The scanner drivers must be current.
  - ISIS libraries support more than 300 scanners. Check www.scannerdrivers.com for a list of ISIS-supported drivers and scanners.
  - If you are running the ImageControls driver, check with Kofax or your scanner manufacturer to ensure you are using the latest release of the scanner drivers.

- ISIS libraries enable full-control support of your scanner's capabilities, allowing you to adjust brightness, contrast, scan resolution, scan mode, dithering, and any other settings, including duplex scanning, that are supported by your scanner

- The speed of the computer is another important factor. ISIS drivers enable scanning at the full rated speed of your scanner. The 32-bit scanning software performs several functions on the CPU. A slow CPU slows this process.

- The speed of the hard disk also affects the speed of the scanner. A slow hard disk will increase the amount of time necessary to write the images to disk.
- The utilization of your network can also cause the scanner performance to degrade. Adding a scanner and creating images on your network can cause packet crashes or bottlenecks. Make certain your network topology and server can function with the increase in network traffic.

Registration Tips and Tricks

Fixing Forms That Fail Registration

The following are possible reasons for forms that have failed registration:

Image Display	Reason and Solution
None or not enough registration fields found	Generally, at least one text registration anchor or two non-text anchors must be found.
	For images that need stretch correction, increase the needed anchors by one.
	If the data is within the field search area, click to show the location of the data and run Auto Tune to modify the registration specification to find the data.
	If the data is outside the field search area, right-click to modify the search area, exit the Registration Anchor dialog, click to show where the data is, and then run Auto Tune.
Enough fields found, but image still did not register.	Generally, each registration field has a maximum amount of x/y shift (from the setup image's answer for the field). If the found data is too far off, or falls below the confidence threshold (set in the Registration Anchor Properties dialog), the data is rejected for registration.
	The data search area must be manually edited by right-clicking in the Registration Anchor dialog.
	There is also a maximum rotation and maximum x/y stretch – these must be manually altered on the Registration Anchor dialog.

Recognition Tips and Tricks

Understanding the Effect of Input Images on Recognition Accuracy

Recognition results are highly dependent upon the quality of the images that are input to the recognition engines. Some factors to consider are:

- Generally, machine print recognition, especially for uppercase and lowercase alphanumeric fields, is significantly better at 300 DPI than at 200 DPI. Internal tests indicate that up to 30% more fields may be rejected on complex machine print forms at 200 DPI vs. 300 DPI. The DPI-dependent results are more pronounced for non-dropout forms.

- Form registration problems caused by improper Image Pre-Processing or Image Enhancement configuration can masquerade as recognition errors. For example, if registration is not highly accurate, mark sense results can be greatly affected. Additionally, improper registration can cause truncation of data from fields, or inclusion of *noise* from adjacent areas. Ensure that registration is accurate before testing recognition on multiple images.

- Whenever possible, avoid fields on forms that have small boilerplate text, or those that have horizontal and vertical lines that need to be removed prior to recognition. This processing is performed by the Image Enhancement client and must be configured and tested on a field-by-field basis.

- Scanner contrast, brightness, bulb color (for form dropout) must all be tuned to maximize image quality. Characters generally are recognized with higher accuracy if they appear to be dark, connected shapes, but not so dark that they are touching one another or have features (such as *holes*) filled in artificially.

- Faxed images are generally of lower resolution and lower quality than scanned images. They may also undergo vertical stretching, which can sometimes be corrected by proper configuration of the Registration zones in Image Enhancement.

Completion Tips and Tricks

Understanding the Effect of the Selective Edits Range Option on Date ADEs

The settings in the Range area of the Field Properties, Selective Edits tab have a significant impact on Date ADE fields within Completion. If you specify a **Range Type** (I-In, O-Out, S-Sel) but do not set range limits (in the **Lower** and **Upper** fields) or put spaces in these limit fields, Completion operators will receive a "Range Error" message during a Completion session. By changing the **Range Type** to "NONE" (i.e., blank), operators will not get this warning message and will be able to key data into the field.

Recovering from Unrecoverable Errors in Distributed Completion

In certain remote processing and data transmission situations within Distributed Completion errors can occur from which recovery is not possible.

This can be particularly troublesome for remote user who may be processing work at all hours of the day or night and cannot recover successfully when the error occurs. In this situation, a System Administrator would have to be available at the host site at odd hours to help the remote user fix the error or processing would stop until the Administrator was available.

In these situations, the following steps must be taken to ensure that information and settings on both the FormWare Server and remote workstation are reset and synchronized before remote processing can continue.

Caution: These steps should be the LAST means of recovery you attempt due to the fact that remotely processed data can be lost.

1. Delete the remote worklist file (RWL) on both the FormWare Server and the remote workstation.

2. Set all batches in the JobFlow to an AVAILABLE state again.

3. Delete any files (BDFs, images, etc.) on the remote workstation that were left behind from the previous attempt.

4. Delete the subdirectory for the FormWare License Manager attached to the remote workstation so that the License Manager will not error out the next time you go back in.

A remote processing recovery option exists to perform these steps from the remote workstation command line. Use the RmtProc /r command line argument to programmatically perform these steps. When initiated with this command line argument, RmtProc will:

- Remove the RWL file on both the FormWare Server and the remote workstation.

- Return all checked out work to the AVAILABLE state in the JobFlow from which it was checked out.

- Remove all temporary files that were created on the client (BDF, images, OCR, batches, etc.).

- Release any licenses held by process.

A confirmation dialog will display when you launch this recovery process to verify that the remote user does not inadvertently perform this process and loose a lot of work.

System Administration Tips and Tricks

Troubleshooting Applications

You can use the following INI parameters to troubleshoot applications at the system, component, or workstation level:

- FWDEBUG - Tests applications and logs debugging information to the error log.

- LOGPATH - Specifies the path to the log file.

- FWERRORCONTEXT - Includes contextual information you specify with each error and debug message.

- FWERROREVENTSOURCE - Specifies how the debug information is logged to the NT event log.

- FWERRORTRAPMSG - Specifies a text string to be captured by the error manager.

Note: To use these INI parameters, you need to be an experienced FormWare application engineer. For assistance, contact Customer Support.

Using Troubleshooting Parameters

You can add these parameters to an existing INI file or create an INI file for this purpose. Place these INI files in the \FormWare\INI\System folder. You can use these parameters in the System.ini, job INI, workstation INI, or a component INI file.

8 *Supplementary Information*

Supplementary Information Overview

This section contains reference information on the various types of parameters that can be used within FormWare to cause the FormWare components to behave in a specific manner. Each parameter has a defined function and can be set in an INI file, within a field in an application, or passed on the command line, depending upon the parameter. The following topics are addressed in this section:

- **INI Parameters** – Parameters you can set within various INI files that interact with FormWare
- **Client Parameters** – Parameters or program switches you can set in the **Client Parameters** field of the JobFlow Task Properties, General tab. You can also set these parameters on a command line.
- **Form ID Parameters** – Form ID training parameters used for Dynamic Form ID. Contains information on each keyword supported by Dynamic Form ID including usage information and the legal parameter limits.

INI Parameters

Parameters are specific settings that apply to one or more FormWare modules and cause the module or application to behave in a specific manner. Each parameter has a defined function and can be set in an INI file, on a tab in the application, or passed on the command line, depending upon the parameter.

In this section, the INI parameters are grouped according to the FormWare component they affect, as follows:

- Administration INI Parameters
- Administration Security INI Parameters
- Completion INI Parameters
- Completion Security INI Parameters
- Recognition INI Parameters
- FileNET Export INI Parameters
- JobFlow INI Parameters
- Miscellaneous INI Parameters

You set INI parameters in INI files, which are text files that control the various functions of the FormWare modules. Each parameter must be placed on a line by itself. All INI files must end with a blank line. However, do not place a blank line between INI parameters.

The INI parameter tables use the following syntax:

- Curly brackets ({ }) indicate choices.
- Pipe bar symbols (|) indicate the boolean OR.
- Angle brackets (< >) indicate that you need to type a certain variable value.
- Ellipses (...) indicate that the value or option can be repeated.
- Square brackets ([ ]) indicate optional items that you may want to include in the parameter setting.

If you want the default behavior, you do not need to include the INI parameter since the default behavior occurs unless specified differently (this has the same affect as specifying N for the parameter unless stated otherwise in the description).

See the Setting INI Parameters section in Designing with FormWare for instructions on how to create and modify INI files.

Administration INI Parameters

The following INI parameters apply to the functionality of the Administration module.

Parameter	Syntax	Description
FPATH	FPATH=<path>	File Path. Defines the directory where the data files will be created and updated.
		Example:
		FPATH=C:\Work\Data
		Location: System.ini, <Job>.ini
FWVBHOST_WAIT_ INTERVAL	FWVBHOST_WAIT_ INTERVAL=1	FormWare VB Host Wait Interval. Delays processing for the defined time in milliseconds (used to correct possible VBA problems associated with dual processor PCs).
		Location: System.ini
FWVBHOST_WAIT_ OVERRIDE	FWVBHOST_WAIT_ OVERRIDE={S	AT}	FormWare VB Host Wait Override. Defines how FormWare handles window focus when using Windows 98/2K/XP.
		Options:
		S - Gives focus to multi threaded processes outside of FormWare.
		AT - Allows FormWare to use the Microsoft Attach Thread patch to define the window that gets focus.
		Location: System.ini
GCA	GCA={Y	N}	Store Network Card address. Store network card address in the first 12 positions of field 22 in the stat record.
		If this option is set, the network card address will also be written to the PROGRAM.LOG file. This option is only active when the data entry is accessed from the application menu.
		Location: System.ini, <Job>.ini, <JobFlow>.ini, <User>.ini, <Group>.ini

Parameter	Syntax	Description
IPP_LOCAL_ IMAGEPATH	IPP_LOCAL_ IMAGEPATH= <path>	IPP Local Image Path. Provides a local image capability to client machines and resolve network problems. Type a path where FormWare should copy images to a local machine. Use if your client machines spend large amounts of time reading images from the network. Location: <Job>.ini
LIMITFILES	LIMITFILES= <max # files>	Limit files. Specifies the maximum number of files displayed in the file list area of the Administration module. Location: System.ini
OPATH	OPATH=<path>	Output Path. Defines the directory that FormWare uses to save OCR files. Location: System.ini
ORD	ORD={Y | N}	On Login Disable. Disables the F4 and F9 keys at login. Location: System.ini
PROGRESSMINIMIZE	PROGRESSMINIMIZE= {Y | N}	Progress Minimize. Automatically minimize the progress box. Location: System.ini, <*Clnt>.ini
PROGRESSPOS	PROGRESSPOS= <x>,<y>	Progress Position. Allows you to position the progress box to your specification of the X and Y coordinates on the screen. Location: <*Clnt>.ini
RMT_BATCHCOUNT	RMT_BATCHCOUNT= <# batches>	Remote Transfer Batch Count. Defines the number of batches to be transferred to remote clients with the Remote Transfer Utility. Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini, Complete.ini
RMT_FORCE_LOGIN	RMT_FORCE_LOGIN= {Y | N}	Remote Transfer Force Login. Requires operators to enter a password when logging into the FW Internet Server. Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini
SHOWALLFILES	SHOWALLFILES= {Y | N}	Show all files. Displays all of the files in a folder, not just the default file types within the Administration module. Location: System.ini
SHOWTHUMBNAIL	SHOWTHUMBNAIL= {Y | N}	Show Thumbnail. Display thumbnail images in all components. Location: System.ini
STRETCH_THRESH	STRETCH_THRESH= <# pixels>	Stretch Threshold. Defines a threshold in pixels to determine if stretch should be applied. Location: System.ini
TCKI_CLIENT_ NORESIZE	TCKI_CLIENT_ NORESIZE={Y | N}	Thin Client Key from Image Client No Resize. When running a thin client key from image session the window resizes to full screen. Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini, Complete.ini

Parameter	Syntax	Description
VBE	VBE={Y | N}	Field selection for OnFieldStart & OnFieldEnd. Defines if fields will trigger OnFieldStart and OnFieldEnd events (Disable both functions with the N value). Location: System.ini, <Job>.ini, <JobFlow>.ini, <User>.ini, <Group>.ini, <*Clnt>.ini
XML_ENCODING	XML_ENCODING= ISO-8859-1	XML Encoding. Encodes ÄÖÜäöüß characters in a format readable by Internet Explorer. Location: XMLClnt.ini
XPATH	XPATH=<path>	OCR Path. Defines a custom path where OCR files are saved. Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini

Administration Security INI Parameters

The following INI parameters determine the security, or accessibility, of the Administration module. To limit accessibility, set the parameter to D (for Disable). When access is denied to a folder, it cannot be opened or viewed. A red $\varnothing$ symbol displays across the folder in the Administration module, indicating that access to that folder is denied.

For example, to deny an operator access to the \FormWare\Devices folder in Administration, add the ACCESS_DEVICES=D parameter to that operators user INI file.

Parameter	Syntax	Description
ACCESS_ADDIMGS	ACCESS_ADDIMGS=D	Add Images mode. Denies access to the ADD IMAGES mode. Location: <User>.ini
ACCESS_BITMAPS	ACCESS_BITMAPS=D	Bitmaps folder. Denies access to the Bitmaps folder. Location: <User>.ini
ACCESS_CHECK_DIGIT	ACCESS_CHECK_DIGIT=D	Check digit folder. Denies access to the Check Digit folder. Location: <User>.ini
ACCESS_COMPLETION	ACCESS_COMPLETION=D	Completion module. Denies access to the Completion module. Location: <User>.ini
ACCESS_DESIGNER	ACCESS_DESIGNER=D	Designer module. Denies access to the Designer module. Location: <User>.ini
ACCESS_DEVICES	ACCESS_DEVICES=D	Devices folder. Denies access to the Devices folder. Location: <User>.ini
ACCESS_DSCAN_ SERVER	ACCESS_DSCAN_SERVER=D	Distributed Scan server. Denies access to the Distributed Scan server. Location: <User>.ini
ACCESS_EDIT_ CURRENT_DIRECTORY	ACCESS_EDIT_CURRENT_ DIRECTORY=D	Edit > Change Current directory menu. Denies access to the Edit > Change Current Directory menu option. Location: <User>.ini

Parameter	Syntax	Description
ACCESS_EDIT_NEW_ ROOT	ACCESS_EDIT_NEW_ ROOT=D	Edit > Add New Root menu. Denies access to the Edit > Add New Root menu option. Location: <User>.ini
ACCESS_EDIT_ PROPERTIES	ACCESS_EDIT_ PROPERTIES=D	Edit > Properties menu. Denies access to the Edit > Properties menu option. Location: <User>.ini
ACCESS_EXPORT	ACCESS_EXPORT=D	Export. Denies access to the Export folder. Location: <User>.ini
ACCESS_EXTERNAL	ACCESS_EXTERNAL=D	External folder. Denies access to the External folder. Location: <User>.ini
ACCESS_FIELD_EDIT	ACCESS_FIELD_EDIT=D	Field edit folder. Denies access to the Field Edit folder. Location: <User>.ini
ACCESS_FILE_EDIT	ACCESS_FILE_EDIT=D	File edit folder. Denies access the File Edit folder. Location: <User>.ini
ACCESS_FILE_NEW	ACCESS_FILE_NEW=D	File > New menu. Denies access to the File > New menu option. Location: <User>.ini
ACCESS_FILES	ACCESS_FILES=D	Files folder. Denies access to the Files folder. Location: <User>.ini
ADDESS_FORMID	ADDESS_FORMID=D	Manual Form ID mode. Denies access to the Manual Form ID mode. Location: <User>.ini
ACCESS_FORMS	ACCESS_FORMS=D	Forms folder. Denies access to the Forms folder. Location: <User>.ini
ACCESS_FWEDIT	ACCESS_FWEDIT=D	Edit programs. Denies access to the FormWare edit program from the Completion module. Location: <User>.ini
ACCESS_FWSCAN	ACCESS_FWSCAN=D	Scan module. Denies access to the Scan module. Location: <User>.ini
ACCESS_IFS	ACCESS_IFS=D	IFS folder. Denies access to the IFS (I-Forms Server) folder. Location: <User>.ini
ACCESS_IFS_CONFIG	ACCESS_IFS_CONFIG=D	IFS\CONFIG option. Denies access to the IFS Config option. Location: <User>.ini
ACCESS_IFS_DATA	ACCESS_IFS_DATA=D	IFS\DATA folder. Denies access to the IFS\Data folder. Location: <User>.ini

Parameter	Syntax	Description
ACCESS_IFS_LOGS	ACCESS_IFS_LOGS=D	IFS\LOGS folder. Denies access to the IFS\Logs folder. Location: <User>.ini
ACCESS_IFS_SCRIPTS	ACCESS_IFS_SCRIPTS=D	IFS\SCRIPTS folder. Denies access to the IFS\Scripts folder. Location: <User>.ini
ACCESS_IMGQC	ACCESS_IMGQC=D	Image QC/Rescan mode. Denies access to the Image QC/Rescan mode. Location: <User>.ini
ACCESS_INDEX	ACCESS_INDEX=D	Index folder. Denies access to the Index folder. Location: <User>.ini
ACCESS_INI_FLDMAP	ACCESS_INI_FLDMAP=D	INI\Fldmap folder. Denies access the INI\Fldmap folder. Location: <User>.ini
ACCESS_INI_GROUP	ACCESS_INI_GROUP=D	INI\Group folder. Denies access to the INI\Group folder. Location: <User>.ini
ACCESS_INI_JOB	ACCESS_INI_JOB=D	INI\Job folder. Denies access to the INI\Job folder. Location: <User>.ini
ACCESS_INI_JOBFLOW	ACCESS_INI_JOBFLOW=D	INI\JobFlow folder. Denies access to the INI\JobFlow folder. Location: <User>.ini
ACCESS_INI_KEYMAP	ACCESS_INI_KEYMAP=D	INI\Keymap folder. Denies access to the INI\Keymap folder. Location: <User>.ini
ACCESS_INI_SYSTEM	ACCESS_INI_SYSTEM=D	INI\System folder. Denies access to the INI\System folder. Location: <User>.ini
ACCESS_INI_USER	ACCESS_INI_USER=D	INI\User folder. Denies access to the INI\User folder. Location: <User>.ini
ACCESS_INI	ACCESS_INI=D	INI folder. Denies access to the INI folder or any of its sub-directories. Location: <User>.ini
ACCESS_JFCLIENT	ACCESS_JFCLIENT=D	JobFlow Client login. Denies access to the JobFlow Client Login dialog. Location: <User>.ini
ACCESS_JFDESIGN	ACCESS_JFDESIGN=D	JobFlow Designer module. Denies access to the JobFlow Designer module. Location: <User>.ini
ACCESS_JOB_FILES	ACCESS_JOB_FILES=D	Job files folder. Denies access to the Job Files folder. Location: <User>.ini
ACCESS_JOB_IMAGES	ACCESS_JOB_IMAGES=D	Job images folder. Denies access to the Job Images folder. Location: <User>.ini

Parameter	Syntax	Description
ACCESS_JOBFLOW	ACCESS_JOBFLOW=D	JobFlow folder. Denies access to the JobFlow folder. Location: <User>.ini
ACCESS_JOBS	ACCESS_JOBS=D	Jobs folder. Denies access to the Jobs folder. Location: <User>.ini
ACCESS_LICENSE_ MONITOR	ACCESS_LICENSE_MONITOR=D	License Monitor. Denies access to the License Monitor. Location: <User>.ini
ACCESS_LISTS	ACCESS_LISTS=D	Lists folder. Denies access to the List folder. Location: <User>.ini
ACCESS_MONITOR	ACCESS_MONITOR=D	JobFlow Monitor. Denies access to the JobFlow Monitor. Location: <User>.ini
ACCESS_OTHER	ACCESS_OTHER=D	Specify Other Command dialog. Denies access to the Tools > Other menu option that opens the Specify Other Command dialog. Location: <User>.ini
ACCESS_PROGRAMS	ACCESS_PROGRAMS=D	Programs folder. Denies access to the Programs folder. Location: <User>.ini
ACCESS_RULES	ACCESS_RULES=D	Rules folder. Denies access to the Rules folder. Location: <User>.ini
ACCESS_SCAN_CFG	ACCESS_SCAN_CFG=D	Scan Cfg folder. Denies access to the Scan Cfg folder. Location: <User>.ini
ACCESS_SORT	ACCESS_SORT=D	Sort folder. Denies access to the Sort folder. Location: <User>.ini
ACCESS_STATISTICS	ACCESS_STATISTICS=D	Statistics folder. Denies access to the Statistics folder. Location: <User>.ini
ACCESS_VALUE_TABLE	ACCESS_VALUE_TABLE=D	Value table folder. Denies access to the Value Table folder. Location: <User>.ini
ACCESS_VBA	ACCESS_VBA=D	VBA folder. Denies access to the VBA folder. Location: <User>.ini

Completion INI Parameters

The following INI parameters are available for use within the Completion module:

Parameter	Syntax	Description
029	029={Y | N}	029 keyboard. Enables data entry with the 029 keyboard, which is the embedded numeric keypad. Replicated as the 029 On option on the Job Properties, Options tab in Designer. Example: 029=Y
10KEY	10KEY={Y | N}	10 Key. Enables the + key on the numeric to act as a field release and the tilde key becomes a +. Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini
ACZ	ACZ=<x>,<y>, <w>,<h>	Adjust character zone. Adjusts the character clip, which displays in the reject repair ticker tape at the top of the Completion window. The values represent the shift in pixels for the zone, i.e., X, Y, W (width), and H (height). These values can be either positive or negative, depending on how you want to change the zone location or if you want to expand or shrink the zone. Location: <Job>.ini
AE	AE=<exception chars>	Alpha Exception Characters. Include this setting to enable international Completion users to type alphabet characters that fall outside the range of A-Z, a-z (e.g., Ä, Ü, Ñ, Æ). Location: System.ini, <Job>.ini
AFE	AFE={Y | N}	Run Edits On Interrupt. Runs the file end edit even when the batch is interrupted. Replicated as the Run edits on interrupt option on the Job Properties, Edits tab in Designer.
AFZ	AFZ={Y | N}	Adjust field zones. Adjusts the field zone highlighting based on the recognition results coordinates. Replicated as the Adjust field zones option on the Job Properties, Image tab in Designer.
ALOC	ALOC={Y | N}	Auto on when location. Turns the auto function on when cursor is positioned on the location field. When the auto function is on, ON appears in the Completion status bar. Replicated as the On when location auto option on the Job Properties, Options tab in Designer.
ALR	ALR={Y | N}	Activates location record in Update mode. If the batch is interrupted during Update mode, this option places the operator at the location record rather than at the beginning of the batch. Replicated as the Location return in update option on the Job Properties, Options tab in Designer.
AOM	AOM={Y | N}	Data will be read in from the OCR file without displaying on the screen. Location: System.ini, <Job>.ini Note: This will bypass any field edits.

Parameter	Syntax	Description
AOVS	AOVS={Y | N}	Oversign after last digit. The oversign key must be pressed after the last character in the field is typed.
		Replicated as the After character option on the Job Properties, Options tab in Designer.
ARE	ARE={Y | N}	Always run edits. Always runs the field edit. Under certain conditions, the field edit may not execute on fields without rejects or fields that never become the location field. This parameter forces the edit to run.
		Replicated as the Always run edits option on the Job Properties, Edits tab in Designer.
ATGL	ATGL={Y | N}	Auto off when back. Turns the auto function off when a back function is selected. The auto functions status displays in the Completion status bar.
		By default, the auto function remains on when the [Page Up] or Record Back key is pressed. If operators need to be able to move back in a batch or data file, this parameter should be set.
		Replicated as the Off when back auto option on the Job Properties, Options tab in Designer.
ATON	ATON={Y | N}	Auto on when forward. Turns the auto function on when a forward function is selected. The auto functions status displays in the Completion status bar.
		If backward movement is allowed in a batch or data file, which requires disabling the auto function, you can use the ATON parameter so that when a [Page Dn] or Record Forward key is pressed, the auto function is turned back on. This enables the operator to continue normal keying (i.e., advance to fields and rejects).
		Replicated as the On when forward auto option on the Job Properties, Options tab in Designer.
BDC	BDC=<column #>	Beginning display column. Enables you to specify a beginning column for displaying fields in Completion. If all the fields are on the left edge of the data form, but you want to display them with a margin in Completion, you can set a beginning display column value.
		Example:
		BDC=20
		Causes all the fields to display in column 20 on the data form, regardless of the position set for the fields in Designer.
		Location: System.ini, <Job>.ini
BIC	BIC=<RGB value>	Background Image Color. Specifies the background image color.
		The 6 characters represent the RGB (red, green, blue) hex values for color. The first 2 characters of each 6 digit string is the Hex value of the red mix, the next 2 are for the green mix, and the last 2 are the blue mix.
		Example:
		BIC=FF0000
		Sets the background color to red.
BKUP	BKUP=<path>	Backup. Defines the directory where backup files will be copied when an operator terminates or interrupts a batch.
		Location: System.ini, <User>.ini, <Group>.ini, <Job>.ini, <JobFlow>.ini

Parameter	Syntax	Description
BLF	BLF={Y	N}	Last field when back. A record back positions the cursor to the last field of the previous form instead of the first.
BOK	BOK={Y	N}	Enable Browse Button. Use this parameter when you have images for a batch that are outside the main batch image folder.
		These outside images can be inserted in Manual IQC mode using one of two methods:
		Enter the full path to the image in the Image Insert Image Name field.
		Enable the Browse button (using BOK=Y) and select an image though the browse dialog.
		Location: <Job>.ini
BRR	BRR={Y	N}	Batch-based reject repair. Enables batch-based reject repair. This displays all reject characters for an entire batch in the reject ticker tape of the Completion module.
		The default behavior is to display the reject characters per record.
		Used with the Batch Build JobFlow task.
		This parameter is replicated as the Batch Reject Repair option on the Job Properties, Recognition tab in Designer.
CAF	CAF={Y	N}	Current attachment first. Enables the current image to be set as the first page in the attachment list. Used with Attachment Processing in Completion.
		Replicated as the Include current image check box on the Job Properties, Image tab in Designer.
CMPLWIN	CMPLWIN= <x>,<y>,<w>,<h>	Configure Completion window. Repositions and resizes the Completion window. The X and Y values determine the X and Y coordinates for the window location, and the W and H values determine the width and height of the window.
		It is primarily used for attachment processing, especially when using the automatic display for attachment images, so you can configure the size and location of the Completion and Image Viewer windows when they display at the same time.
		Location: System.ini
CRI	CRI={Y	N}	Combine reject ticker and Image Viewport. Provides uniformity in how the data form is displayed in the Completion window by combining the reject ticker tape and the image viewport into a single window.
		When Completion switches from a mode displaying the ticker tape to one that doesn't, the space taken by the ticker tape is given to the image viewport.
		Default: Y
		If you do not set this parameter to Y, the space taken by the ticker tape will be given to the data form, changing the number of data fields viewed on the data form from one mode of Completion to the next.

Parameter	Syntax	Description
CURL	CURL=<RGB value>	Cursor Color. Change cursor to the defined color. The 6 characters represent the RGB (red, green, blue) hex values for color. The first 2 characters of each 6 digit string is the Hex value of the red mix, the next 2 are for the green mix, and the last 2 are the blue mix. Example: CURL=FF0000 Sets the color to red.
CURSZ	CURSZ=<number>	Cursor Size. Change the size of the cursor. Example: CURSZ=30
DAF	DAF={Y | N}	Automatically display attachments. Displays attachments in the Image Viewer for each record containing attachment images without a Completion operator having to select a menu item to view them. An attachment image is one that is identified with an attachment form type and is listed after a record image in the BDF. Location: System.ini, <Job>.ini
DEF	DEF={Y | N}	Destructive record reject. Enables the destructive record reject function, which places a pound character (#) in the first position of the field when it is rejected, and a reject code in the second position of the field when the record is rejected. This destroys the field data in the first and second positions. The record reject code is determined by the operators selection in the record reject dialog.
DEI	DEI={Y | N}	Display enhanced image. Specifies the enhanced image to be the default image displayed in Completion.
DFC	DFC={Y | N}	Display Field Clips. Changes the viewport to the masked zone type so the operator can only see the field zone, not any of the surrounding text. Also disables the full screen image viewer and image scroll functions. Required for remote processing.
DFL	DFL={Y | N}	Dynamic form loading. Loads one data form at a time in Completion instead of the default behavior of loading all data forms in the job at transaction start. This parameter is optimal for transaction processing or jobs with a large number of forms as it minimizes time on startup. Forms for a job will only be loaded into memory when the forms are used. This can decrease the amount of time to load an application into memory when using Completion or Designer. Location: System.ini, <Job>.ini Caution: Before using this parameter, make certain that all data forms in the job will not cause failure or error in a production environment. Otherwise, Completion will display an error message and shutdown upon encountering a data form causing an error.

Parameter	Syntax	Description
DIO	DIO={Y | N}	Delete Form. Defines if operators are allowed to delete a record from a batch in Completion.
		Enables an operator to delete a record that has an image.
		Replicated as the Delete image records option on the Job Properties, Image tab in Designer.
		Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini
DKYS	DKYS=<key number>[,...]	Disable keys. Disables the specified key and menu option during a Completion session. Each key must be three characters. To disable multiple keys, separate each key by a comma.
		Refer to the Mapping Keys section for a list of these values.
		Example:
		DKYS=297
		Disables both the Record Delete key and its menu item.
		Location: System.ini, <User>.ini, <Group>.ini, <Job>.ini, <Jobflow>.ini
DLE	DLE={Y | N}	Display all recognition data. Displays all recognition data in Entry mode, but enables blind verification. The operator is positioned on the first reject.
		Replicated as the Recognition data option on the Job Properties, Display tab in Designer.
DMU	DMU=<Menu name>:[<Menu option>[;...]]	Disable menus and menu items in Completion. Disables display of menus and menu items within Completion. This parameter is applied on Completion session startup and can be used at the system level or task level.
		The system level approach affects all Completion sessions on the system.
		The task level approach affects specific Completion tasks. To do this, you must create a unique INI file (in the FormWare\Ini\System folder) that contains the DMU parameter references for the menus and menu items you want to disable for that Completion task.
		You can specify multiple DMU properties in the INI files. Each DMU line is used to disable display of either individual menu items on a pull-down menu or an entire menu from the Completion window menu bar. The first value to the right of the equal sign is the name of the menu and is delimited by a colon. Subsequent values (delimited by semi-colons) are the names of menu items on that pull-down menu.
		Each DMU parameter must be separated by a <crlf> (carriage return/line feed combination) within the INI files.
		Examples:
		DMU=File:Append;Verify;Update
		Disables display of the three menu item selections (Append, Verify, and Update) from the File menu. (The File menu still displays on the menu bar in this example.)
		DMU=View:
		Disables display of the View menu from the menu bar.
		Location: System.ini, <UniqueTask>.ini, INI=<filename>.ini command line property

Parameter	Syntax	Description
DRM	DRM={Y | N}	Display reject message. Displays reject messages in the status bar of Completion. While the cursor is positioned on the current field where DRM is set, Completion looks in the BDF for the FW_REJ_# property, where # is the current field number. You can use this parameter with the FLDREJS=Y parameter and value that tells the recognition engine and the rule client to write the field reject reason to the BDF. Location: System.ini, Complete.ini
DSC	DSC={Y | N}	Define Space Character. Forces rejected fields to be space filled. Location: System.ini, <Job>.ini
EAU	EAU={Y | N}	Auto on immediately. Turns the auto function on immediately. The auto functions status displays in the Completion status bar.
EFJ	EFJ={Y | N}	Field justification. Performs justification of the field data when the current record key is pressed.
EMPTY_FLD_ COORDS	EMPTY_FLD_ COORDS={Y | N}	Empty Field Coordinates. Displayed clips in Completion will be the same as in FormWare 4.6x if the value is set to N. Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini
ENH_IMAGE_ NO_WRITE	ENH_IMAGE_NO_ WRITE={Y | N}	Disable Enhanced image creation. Disables the creation of enhanced images for pages identified as attachments. Location: Enhclnt.ini
EOF	EOF=<Edit Key #>	End Of File. Calls the OnEditKey event after OnTransactionEnd event to close files currently processing in Completion. This allows you to issue an edit key after onTransactionEnd and before OnProjectEnd to add KFP (Key From Paper) batches into the JobFlow. Location: System.ini, <Job>.ini
ERL	ERL={Y | N}	End of Record Release. Takes the operator to the end of a record before releasing to the next record. Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini
FEFG	FEFG={Y | N}	Field edit on flag. Forces the field edit to execute when an operator flags a field.
FES	FES={Y | N}	Force Edit Field Start. Runs the start OnFieldStart routine on the last record when you have backed up and are now moving forward with a page down. Location: System.ini, <Job>.ini
FLDREJS	FLDREJS={Y | N}	Filed reject reason. Tells the recognition engine and the rule client to write the field reject reason to the BDF. This allows the rules reject reason to be accessed during Completion. Used with the DRM (display reject reason) parameter that displays the reject reason in the status bar during Completion. Location: Recognition.ini
FOC	FOC={Y | N}	Field rejects to flags. During batch build, changes field rejects to flags so they can be repaired in Remove Flags mode. You can use this parameter with multi-pass JobFlow systems so you can key field rejects as a separate pass. Replicated as the Separate chr/fld repair option on the Job Properties, Image tab in Designer.

Parameter	Syntax	Description
FRP	FRP={Y | N}	Field release on position. Forces a field release on a position statement during a field edit. The parameter disables any movement in the data file and forces location position for continuous keying.
FSE	FSE={Y | N}	Force Start Edit. Forces a start field edit on the last record even after an operator has backed up and then moved forward (i.e., pressed the [Pg Down] key). A field edit is designated as a start field edit when you select the S-Start item for the Field Edit option on the Field Properties, Selective Edits tab. Location: System.ini, <Job>.ini
FTC	FTC={Y | N}	Field type checking. Enables field type checking for reject repair. Compares recognition data against the field type, and converts any character that does not match to a character reject. Recognition results can return a value not of the field type set for a field, but places the data in the field without error. Set this parameter to force a check of the data. Replicated as the Perform type checking option on the Job Properties, Edits tab in Designer. Examples: FTC=N (or FTC not set) If an unsigned numeric field receives results of 209F44568, the data is accepted. FTC=Y If an unsigned numeric field receives results of 209F44568, the data result is 209?44568.
FWRESULTS	FWRESULTS= {B | I | S | T | M} [...]	FormWare Results. Logs result values as work is being processed. Options: B - Log results for each batch I - Log results for each image S - Log status indication for each batch or image ("ERROR" or "OK") T - Log timing value for each batch or image M - Log current memory usage for each batch or image Note: Multiple values are allowed. Files are generated in the following folders: <Jobflow> FormWare/Jobflow <User> FormWare/Tmp/Diagnose Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini, <*Clnt>.ini

Parameter	Syntax	Description
FWSKIP ONREJECT	FWSKIPONREJECT= <String>	FormWare Skip Reject. Components other than Completion will skip records that have been rejected with the specified reasons.
		The string value can be one or more reject code characters.
		When this parameter is invoked and a previously rejected record has a reject code that matches one of the characters in this string, the rejected record is skipped. If the asterisk character (*) appears in the string, then any reject code character will cause the rejected record to be skipped.
		Location: System.ini, <Job>.ini, <JobFlow>.ini, <*Clnt>.ini
GCHR	GCHR={Y | N}	Graphic Characters. Allows graphic characters to be keyed. This must be used in conjunction with the field type-mapping feature to remap keys for a mode to graphic keys.
		Location: System.ini
HARDCOPY	HARDCOPY={Y | N}	Hardcopy documents required. Indicates that the current batch does not contain images, enabling users to process Paper Only batches through JobFlow.
		Displays a Batch <batchname> requires hardcopy documents message to the operator when opening a batch assigned this parameter.
ICS	ICS={(Y | N) | ((Y | N), (Y | N)) | <path to directory>}	Intelligent caching system. Provides a system of caching by job.
		Options:
		Y - Use just the Y value to set the Internal Read Cache. This is optimal for transaction processing.
		Y,Y - Use the Y,Y values to add the Internal Write Cache to the Internal Read Cache.
		<path> - Use the <path to directory> to indicate an existing directory to copy application files to the local machine for caching.
		Examples:
		ICS=Y
		ICS=Y,Y
		ICS=C:\Work\Cache\
		Location: System.ini, <Job>.ini, Completion.ini
IDR	IDR={Y | N}	ID required. If an image assigned a form number 99 is loaded into Completion, Completion automatically switches into Identify Images mode for the operator to designate the correct data form type.
ILB	ILB=<x>,<y>,<w>,<h>	Initialize list box. Resizes and repositions the window for a list box, display box, or pick list instead of the windows default position in the center of the screen. The X and Y values determine the X and Y coordinates for the window location, and the W and H values determine the width and height of the window.
		Generated by FPL, a pick list window displays in Completion when operators access an index value for which there is more than one match (i.e., Smith, John on Elm Street and Smith, John on State Street).

Parameter	Syntax	Description
IMAGES	IMAGES=<path>	Images. The Job Images folder displays all images. An image is a digital representation of a paper form or document. All image files within the FormWare system must be stored as Group 4 TIFF files for a job.
		You must specify the IMAGES=<path> parameter in the job INI file to display the images for a specific job.
		The path must be the full path to the images directory, including the drive letter. When you select the Job Images folder on the File Browser, the Job Selection dialog displays. Select a job from the list and select the OK button to display all of the image files in the directory specified for that job. If you want to display the images for all of the jobs, do not select a job and select the OK button. All of the functions available in the Images folder are available in the Job Images folder.
		You can also use this parameter to specify the directory where batch images are saved in the BDF file. If this value is used the path to the images does not need to be entered for every page.
		Location: <Job>.ini, <Batch>.bdf
INI	INI=<INI file>	Completion startup INI. A command-line property that specifies an INI file in the FormWare\Ini\System folder. The INI file that it references contains the parameters that normally go on the command line as startup parameters to Completion. The property is especially useful in conjunction with the DMU parameter (to facilitate the deletion of Completion pull-down menu items without having to type lengthy command-line parameters). Refer also to DMU.
IRR	IRR={Y	N}	Images required. Images are required for all records.
		Replicated as the Only allow image records option on the Job Properties, Image tab in Designer.
ITQ	ITQ=<# of seconds>	Interrupt Time. The time before Completion will automatically close and exit, given in seconds.
		Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini
LARGEFONT	LARGEFONT= {Y	N}	Large Font Enable large font support.
		Location: System.ini
LFC	LFC={Y	N}	Link format correct. In an FPL field edit, the LINK verb performs a format correct on the current record rather than linking on a record release.
		By default, the LINK verb is not invoked until the current record is released. For example, if the operator is on field 3 and the field edit directs the LINK to format 4, the operator will not see the change until keying for all fields in the record is complete. If LFC is set to Y, the data form changes immediately.
		Replicated as the Link performs fmtcor option on the Job Properties, Options tab in Designer.
LNB	LNB={Y	N}	Load next batch. Terminates the batch automatically when all images are processed, and if using JobFlow, loads the next batch.
MFS	MFS={Y	N}	Master File Synchronization. Used as a maintenance program for master files built for FormWare applications that synchronize Completion and the PROTECT and CLEAR verbs.
		Location: <Job>.ini

Parameter	Syntax	Description
MINIMIZE	MINIMIZE={F | M | P}	Window size.
		Options:
		F – Full screen window. The Completion window remains at full screen between batches and does not minimize.
		M – Minimize window. The Completion window is minimized at the start of the module. To display the window, access it from the task bar.
		P – Pop-up window. The Completion window displays as a pop-up, maximizing and minimizing with each batch or transaction processed.
		Default: P
NBL	NBL={Y | N}	No Batch Lock. Use this as an environment variable to make a local copy of a batch data file. Usually eliminates Unlocking err -1 and locking error 15 in Completion. Copies the batch data file to the server when the batch is interrupted or terminated.
		Creates the batch data file on the local machine. Copies the batch data file to the server when the batch is interrupted or terminated.
NCRW	NCRW={Y | N}	After the operator has performed a record display function in verify mode, do not clear the screen once they have resumed keying data.
		Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini, <*Clnt>.ini
NDI	NDI={Y | N}	Hide image display. Disables the image viewport display.
		Replicated as the Hide image display option on the Job Properties, Display tab in Designer.
NE	NE=<String>	Numeric Exception. Specify up to 9 numeric exception characters.
		Location: System.ini, <Job>.ini
NEP	NEP={Y | N}	New event processing. Implemented to create more consistency in Completion event handling.
		Provides access to new Image and Attachment VBA events added to Completion thereby maintaining backward compatibility.
		NEP=Y is the default for FormWare Database. It can be disabled by placing a NEP=N in the System.ini file.
		NEP=Y should be used with all new FormWare applications, and must be set when using Completion in FormWare File System.
		Function Timing:
		OnTransactionStart
		OnDocumentStart
		OnImageStart Image events not fired before
		OnFormStart Fires after image and ocrdata are loaded
		OnFieldStart Fires on non-display fields
		Transaction close
		OnFieldEnd Fires on non-display fields
		OnFormEnd
		OnImageEnd Images events not fired before

Parameter	Syntax	Description
		OnDocumentEnd
		OnTransactionEnd
		Interrupt
		OnTransactionEnd
		Attachment Events
		OnAttachmentStart
		OnAttachmentEnd
		Only in Completion
		Only with Internal Attachment Viewer
		Location: System.ini, Complete.ini
NKYS	NKYS=<function key>	Net Key Strokes. Specifies up to 15 function keys that will be included in the net keystrokes rather than the gross keystrokes.
		Location: System.ini, <Job>.ini
NOFL	NOFL={Y | N}	No Log File. Do not display the file log if is present.
		Do not display batch log. If selected, displays the batch log to the operator.
NOKSMACROS	NOKSMACROS= {Y | N}	No Key Stroke Macros. Defines a keystroke macro to automatically stop for the input and accept statements.
		Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini
NOO	NOO={Y | N}	No Operator Output. Disables the operator statistics file.
		Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini
NOOP	NOOP={Y | N}	Disable operator statistics. Disables operator statistics for particular jobs, such as ones that are difficult to key or one-time jobs.
		Replicated as the Operator option on the Job Properties, Statistics tab in Designer.
NOVR	NOVR={Y | N}	Disable oversign release. Does not release a field when the oversign key [F8] is pressed enabling users to have multiple oversign characters per field.
		Replicated as the No release oversign option on the Job Properties, Options tab in Designer.
NRI	NRI={Y | N}	Normal record insert in Update. Performs location field functions when a record is inserted in Update mode. The default behavior does not enable location field functions when an operator inserts a record in Update mode.
		Replicated as the Record insert in update option on the Job Properties, Options tab in Designer.
NRS	NRS={Y | N}	No record synchronization. Does not perform record synchronization. The cursor positions to the record moved to by a field edit. Set this parameter to physically show a previous record when the BACK verb is used.
		Location: System.ini, <Job>.ini

Parameter	Syntax	Description
NTC	NTC={Y | N}	New type checking. Redefines the Selective Edits functionality to include the fields default character set and the specified characters. Allows an operator to enter Alpha Only or Numeric field type, plus whatever you type in the Include field.
		By default, if you select an Alpha Only field type and in the Include option you type "YN", an operator would only be allowed to enter Y or N.
		If you set NTC=Y, select an unsigned numeric field type, and in the Include option you type "YN", an operator would be allowed to enter numeric characters and also Y and N.
		Replicated as the Include field on the Field Properties, Selective Edits tab in Designer.
		Location: System.ini, <Job>.ini
OMR	OMR={Y | N}	Mark sense verification. Enables Optical Mark Recognition (OMR) verification.
OVI	OVI={Y | N}	Overlay Image. Allow an overlay image to be used in completion.
OVO	OVO=<character>	Oversign Character. Specifies the oversign character.
		Default: &
		Some mainframes use a value other than an ampersand (&) to denote a negative number ending in 0.
		Replicated as the 0 character field in the Oversign Options area on the Options tab of the Job Properties window in Designer.
		Location: System.ini, <Job>.ini
PRN	PRN={Y | N}	Allow the operator to print a form Allows the operator to print a form.
		Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini, Complete.ini
PRT	PRT={Y | N}	Enable printing. Enables an operator to print an image from the Completion module.
QCO	QCO={Y | N}	Quality Check rejects only. When opening a batch in Quality Check (IQC) mode, the operator verifies only those fields that had rejects repaired in a previous pass. All fields used with this parameter must have the Verify option set to Visual on the Field Properties, Checkbox Edits tab within Designer.
		The QCR parameter must also be set to indicate on which records to stop.
QCR	QCR={<record number>[,...] | <record number>R}	Quality Check records. Specifies the records displayed in Quality Check (IQC) mode. Type either a number separated by a comma or a number followed by an R for every nth record. This defines which records are advanced to in Quality Check mode.
		Examples:
		QCR=3,15,20,31,45
		Only shows those 5 records for verification
		QCR=3R
		Shows every third record

Parameter	Syntax	Description
RFR	RFR={Y | N}	Remove flags required. In Remove Flags mode, requires all flags to be removed before an operator can terminate the batch.
		Replicated as the Replay field flags option on the Job Properties, Options tab in Designer.
RNFE	RNFE={Y | N}	Release No Field Edit. Disables the Field level edits when a field is released.
		Does not call an FPL or VBA field edit when a field is released.
		Location: System.ini, <Job>.ini, Complete.ini
RPD	RPD={Y | N}	Read page data. Reads data from the BDF. Up to 4000 characters can be read from a BDF per image. The data such as a document tracking number or a claim number may not appear on the data form, but is associated with the image and can be extracted from the BDF. This data can be placed in the BDF by the Scan module or programmatically.
		Replicated as the Read page data option on the Job Properties, Recognition tab in Designer.
SAF	SAF={Y | N}	Skip attachment form. Does not load, decompress, nor display attachment images during a Completion session. The Image Viewer can display images, but no data record is created.
SAV	SAV={Y | N}	Save image. Allow the operator to save a form.
		Enables the operator to save a form after image manipulations in Completion.
		Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini, Complete.ini
SBP	SBP={Y | N | S}	Scroll by page.
		Options:
		Y - Scroll by page. Scrolls the entire data form a page at a time if not all of the fields are visible. The Scroll by page option directs Completion to calculate the most efficient way to scroll the data form to minimize the number of scrolls performed as the operator keys data. Replicated as the Scroll by page option on the Job Properties, Display tab in Designer.
		N - Scroll a line at a time.
		S - Scroll By Sub-Page. Controls which fields display during scrolling. You can use this option instead of Y to enable the FormWare application designer to determine when and how scrolling takes place. Use the ^ as the first character in a label to indicate the beginning of a sub-page.
		Default: N.
		Location: <Job>.ini
SCN	SCN={Y | N}	Enable scan. Enables an operator to rescan an image, overwriting the current image with the newly scanned one. The workstation must have a scanner attached.
		Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini, Complete.ini

Parameter	Syntax	Description
SECURITY TIMEOUT	SECURITYTIMEOUT= <# seconds>	Security Time Out. Defines the delay time a client will wait to receive a license from the Security License Manager.
		Use this property if a client gets the "Security check timeout" error message.
		Default: 10
		Location: System.ini, <Job>.ini, <JobFlow>.ini, <User>.ini, <Group>.ini
SEK	SEK={Y | N}	Skip edits and Force FPL Edit Key before VBA OnEditKey. The checkbox edits are skipped when a Field Edit key is pressed. Forces the FPL edit key logic to process before the VBA OnEditKey function.
		Used when edit key logic is called from within a VBA program (i.e., OnEditKey event). The default behavior is that when FLDEDITKEY logic within FPL is called, control is always maintained by the VBA program. Because control is never passed to the FPL program, any edit key logic in the FPL program is ignored. Other FPL code executes as normal.
		Location: System.ini, <Job>.ini, <JobFlow>.ini
SFD	SFD={Y | N}	Single Field Display. Display only one field at a time.
		Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini, Complete.ini
SNA	SNA={Y | N}	Spaces not allowed. Does not allow spaces in the field data.
STG	STG={Y | N}	Scale to Gray. Enables the Completion operator to zoom in or scale an image to a significant degree and retain image quality using scale to gray functionality.
		Location: System.ini, <Job>.ini
TAO	TAO={Y | N}	Toggle auto on. Leaves the Auto function active when errors occur. Without this parameter set to Y, the error message causes the auto function to toggle off. If Auto is off, the operator cannot automatically advance to the next field with rejects.
		Replicated as the On when error auto option on the Job Properties, Options tab in Designer.
TEI	TEI={Y | N}	Transfer Enhanced Image. Sends the enhanced image to distributed Completion operators.
		Location: System.ini, <Job>.ini
TIO	TIO={Y | N}	Terminate incomplete batch. Enables the operator to close an incomplete batch and force an edit to run on an application-specific level. Use this parameter with great caution. It can lead to exporting partial data files.
		Replicated as the Close at anytime option on the Job Properties, Options tab in Designer.
TNFE	TNFE={Y | N}	No field edit on tab. Does not call an FPL or VBA field edit when the [Tab] key is pressed. The [Tab] key by default acts as a field release, which executes the edit. With this parameter, you can use the [Tab] key to bypass rather than release.
		Location: Complete.ini
TWV	TWV={Y | N}	Terminate with verify. Automatically terminates a batch in Verify mode when the last record has been verified.
		Replicated as the Auto close in Verify option on the Job Properties, Options tab in Designer.

Parameter	Syntax	Description
UFAT	UFAT={Y | N}	Write to disk on release. The batch file is updated when the record is released. If you do not set this option, several records are stored in memory (depending on available memory) before updating the batch file. This option may increase network traffic. Use this parameter if power outages are common or very large batches are written, to prevent the loss of data. Location: Complete.ini
UHR	UHR={Y | N}	Use header for image information. Checks for flags, such as rotation, in the TIFF image header. By default, the Completion module ignores the orientation flag in the TIFF header. Replicated as the Orientation from image option on the Job Properties, Image tab in Designer.
USERNAME	USERNAME=<String>	User Name. Used to change a username for operator stats, etc. Must be placed in environment. Also must be running Completion from command line for this to work.
VBALOCAL	VBALOCAL={Y[A] | N}	Copy VBA project locally. Copies the completion VBA project to the local workstation. Options: Y - Downloads the VBA project when a new project is needed. A - Checks the date on the local version. If the server version is newer it is copied to the local workstation. Examples: VBALOCAL=YA VBALOCAL=Y Location: System.ini, environment variable
VFF	VFF={Y | N}	Verify field form properties. In a VBA OnTransactionStart event, use FW.SetProperty ("VFF=Y") to force Completion to shut down if it cannot open the VBA project. Location: <Job>.ini
VIEWWIN	VIEWWIN= <x>,<y>,<w>,<h>	Configure Image Viewer window. Repositions and resizes the Image Viewer window. The X and Y values determine the X and Y coordinates for the window location, and the W and H values determine the width and height of the window. When using the automatic display for attachment images, this parameter enables you to configure the size and location of the Completion and Image Viewer windows when they appear on the screen at the same time. Location: System.ini
VNE	VNE={Y | N}	Verify not entry operator. Ensures that the operator who entered the batch cannot access it in Verify mode. Use this parameter when you do not want an entry operator to verify his own work.
VRT	VRT={Y | N}	Verify requires terminate. Does not allow verification to be performed on an incomplete batch.

Parameter	Syntax	Description
VSM	VSM=<integer>	Multiply Attachment Viewer Scroll Value. The Attachment Viewers scroll option is set in Designer on the Job Properties, Image tab. The maximum value accepted in the Scroll field is 100 pixels. The VSM parameter lets you set scroll values greater than 100.
		The integer specifies the number by which you want the Scroll field value to be multiplied.
		The VSM parameter only applies when the embedded Attachment Viewer is used to display attachments in Completion.
		Example:
		VSM=3
		If the Scroll value is 75 in Designer and VSM equals 3, the attachment scrolls by 75 x 3 = 225 pixels.
		Location: System.ini, <Job>.ini
XFA	XFA={Y	N}	Extended field alignment. Used by Completion only to attempt to display an I-form created with tables the same way that it would appear in a browser.
		Location: <Job>.ini
XFC	XFC={Y	N}	Form change after record back. Provides the ability to do a form change [F5]-[F9] after a record back.
		Location: System.ini, <Job>.ini
ZEK	ZEK=530	Zone Editkey. Triggers an EditKey event in VBA when a user draws a rectangle on the image using the mouse.
		Location: <Job>.ini

Note: If a parameter requires a directory path, you can replace part of the path with the %FORMWARE% argument. When you use this argument, the software substitutes the value specified by the FORMWARE environment variable.

Completion Security INI Parameters

You use the ACCESS_<mode>= INI parameter to determine the security of the Completion module. To grant access, set the parameter to E for Enable. To deny access, set the parameter to D for Disable. For example, to deny an operator access to the New mode of Completion, add the ACCESS_DENEW=D parameter to that operators user INI file.

The following table lists the parameters for Completion security:

Parameter	Syntax	Description
ACCESS_APPEND	ACCESS_APPEND={E	D}	Append mode. The D value denies access to the Append mode of Completion.
ACCESS_CORRECT	ACCESS_CORRECT={E	D}	Correct mode. The D value denies access to the Correct mode of Completion.
ACCESS_DE	ACCESS_DE={E	D}	Completion module. The D value denies access to the Completion module.
ACCESS_DENEW	ACCESS_DENEW={E	D}	New mode. The D value denies access to the New mode of Completion.
ACCESS_QUALCHK	ACCESS_QUALCHK={E	D}	Quality Check mode. The D value denies access to the Quality Check mode of Completion.

Parameter	Syntax	Description
ACCESS_READONLY	ACCESS_READONLY={E | D}	Read Only mode. The D value denies access to the Read Only mode of Completion.
ACCESS_REMFLGS	ACCESS_REMFLGS={E | D}	Remove Flags mode. The D value denies access to the Remove Flags mode of Completion.
ACCESS_UPDATE	ACCESS_UPDATE={E | D}	Update mode. The D value denies access to the Update mode of Completion.
ACCESS_VERIFY	ACCESS_VERIFY={E | D}	Verify mode. The D value denies access to the Verify mode of Completion.

Recognition INI Parameters

The following INI parameters apply to the recognition process.

Parameter	Syntax	Description
ENHANCE_ DOTMATRIX	ENHANCE_DOTMATRIX= {Y | N}	Enhance Dot matrix. Used to disable dot-matrix detection in the Inscript2 engine. Used in conjunction with FORCE_DOTMATRIX_TO_MTXT. This is needed on certain invoice type images with multiple fonts that get "oversmeared by and produce garbage recognition. You need to turn these settings on in the INI file to get good recognition on these types of images. Pages detected as dot matrix will be darkened and smoothed in Inscript2. Default: Y Location: In2Clnt.ini
FORCE_DOTMATRIX	FORCE_DOTMATRIX= {Y | N}	Force Dot matrix. Forces machine text engine for images it detected as dot matrix. Default: Y Location: System.ini, <Job>.ini, In2Clnt.ini
FORCE_ DOTMATRIX_ TO_MTXT	FORCE_DOTMATRIX_ TO_MTXT={Y | N}	Force Dot matrix to mtext. Used to disable dot-matrix detection in the Inscript2 engine. Used in conjunction with ENHANCE_DOTMATRIX. This is needed on certain invoice type images with multiple fonts that get "oversmeared by and produce garbage recognition. You need to turn these settings on in the INI file to get good recognition on these types of images. Pages detected as dot matrix will be darkened and smoothed in Inscript2. Default: Y Location: In2Clnt.ini
FW_INS_INSRAW	FW_INS_INSRAW={1 | 0}	Saves raw recognition data. Saves the raw recognition data to FormWare\TMP\Diagnose\Ins Location: <Recognition>.ini
FW_INS_NO_ REMOVALS	FW_INS_NO_REMOVALS= {Y | N}	Inscript No Removals. Do NOT run removals on Handprint fields. Location: InsClnt.ini

Parameter	Syntax	Description
FWSLM_OPTION	FWSLM_OPTION=<String>	FormWare SLM Option. Defines the licensed speed used for a specific recognition engine (The speed specified must be a licensed recognition speed). Location: <Recognition> .ini
KEEP_VERTICAL_ SPACING	KEEP_VERTICAL_ SPACING={Y | N}	Keep Vertical Spacing. Used in InScript to maintain data integrity in columns. Location: Inscript.ini

FileNET Export INI Parameters

The following INI parameters apply to the FileNET Export component, which are set in the job INI file.

Parameter	Syntax	Description
FNDOCCLASS	FNDOCCLASS= {Y | N}	FileNET document class. Specifies the name of the FileNET document class for which batches are created. This parameter is typically specified in the job INI file because different jobs should be associated with different document classes. If you do not set this parameter, the name in the Document Name field on the Form Properties, Edits tab specifies the document class. Location: <Job>.ini
FNERRPHASE	FNERRPHASE= {INDEX | VERIFY | COMMIT | COMMIT_ASYNC | COMMIT_ASYNC | DELETE}	FileNET phase error. Specifies the disposition of a non-successful batch after the FileNET Export component is finished. Batches can be left in the BES phase, committed to the IMS phase, or deleted. Options: Refer to the next table.
FNNBO	FNNBO={Y | N}	FileNET no busy override. Disables the busy override processing of a batch. If busy override is turned off, batches with a Busy status are not processed by the FileNET Export component. The default behavior overrides busy batches and processes them.
FNPHASE	FNPHASE= {INDEX | VERIFY | COMMIT | COMMIT_ASYNC | COMMIT_ASYNC | DELETE}	FileNET phase successful. Specifies the disposition of a successful batch after the FileNET Export component is finished. Batches can be left in the BES phase, committed to the IMS phase, or deleted. Options: Refer to the next table.

The values for the FNPHASE and FNERRPHASE parameters are the same and are described in the following table.

Value	Description
INDEX	Index phase. Assigns the Index phase as the next phase for the batch.
VERIFY	IndexVerify phase. Assigns the IndexVerify phase as the next phase for the batch.
COMMIT	Commit phase. Assigns the Commit phase as the next phase for the batch. This is the default phase when the FNPHASE parameter is not assigned a specific value.

Value	Description
COMMIT_ASYNC	Asynchronous batch committal. Performs asynchronous batch committal. The FileNET Export component places the batch in the Commit queue and then processes another batch. Successfully committed batches are automatically deleted from the BES. This is the recommended setting for committing batches.
COMMIT_SYNC	Synchronous batch committal. Performs synchronous batch committal. The FileNET Export component waits until the batch is successfully committed before processing another batch. Successfully committed batches are automatically deleted from the BES. Due to the length of time it can take to completely commit a batch, this setting is not recommended.
DELETE	Batch delete. Deletes the batch from the BES. Do not use this setting except while testing an application or for batches that have errors.

JobFlow INI Parameters

The following INI parameters apply to the JobFlow Client and JobFlow tasks.

Parameter	Syntax	Description
ALLOWJOBFLOW	ALLOWJOBFLOW= <jobflow name>:<task #>[,...]	JobFlow restrictions. This option controls the JobFlow Client tasks. It does this by limiting the JobFlow Client only to those JobFlows and JobFlow tasks listed in the INI file. These values are not case sensitive. Example: ALLOWJOBFLOW=JobFlow1:1,3,8 Location: <User>.ini, <Group>.ini
FWERRORBEEP	FWERRORBEEP= <freq>,<duration>,<interval>	JobFlow Client. Defines the duration, intensity, and frequency of the sound played when an error message displays during processing. By default, turning ON the Run Minimized function turns OFF the Error Beep function for common error dialogs. If you want to hear the beep for tasks that have the Run Minimized function turned on, you must add this INI file parameter to the appropriate INI file. Options: freq - The frequency of the beep tone in Hz (Range 37 - 32767, suggested value 22050). duration - The duration of the beep in milliseconds (must be greater than zero, suggested value 500). interval - The interval value is how often the beep should occur in milliseconds (must be greater than the value for duration, suggested value 2000). Location: System.ini, <Jobflow>.ini

Parameter	Syntax	Description
JFCMAXRUNHRS	JFCMAXRUNHRS= <# hours>	JobFlow Client Max Runtime Hours. The JobFlow Client restarts at the same task after the specified number of hours. Once the maximum is reached, the restart occurs without any manual intervention. Range: 1-10,000
JFCMAXRUNMINS	JFCMAXRUNMINS= <# minutes>	JobFlow Client Max Runtime Minutes. The JobFlow Client restarts at the same task after the specified number of minutes. Once the maximum is reached, the restart occurs without any manual intervention. Range: 1-600,000
JFD	JFD=<# seconds>	Job Flow Delay. Defines the number of seconds before JOBFLOW servers request a new batch. The default is 2 seconds. Note: This value may be set higher on networks experiencing heavy traffic to reduce collisions. Location: System.ini, <Job>.ini, <JobFlow>.ini
JFT	JFT=<# seconds>	Job Flow Terminate. Defines the amount of time in seconds an error will be displayed on screen when using JobFlow with image pre-processing or recognition. Location: System.ini, IPPClnt.ini, Complete.ini
JNF	JNF={Y | N}	Job Name Files. Prompt the user with a Formware Open File dialog box to prevent the possibility of operators deleting files. Location: System.ini, <Job>.ini, <User>.ini, <Group>.ini
NFD	NFD={Y | N}	New File Dialog. Prompt the user with a FormWare Open File dialog box to prevent the possibility of operators deleting files using the standard windows dialog box.
NOEXCEPTIONHANDLING	NOEXCEPTIONHANDLING= {Y | N}	JobFlow Client. Disables the try/catch logic, which detects derived unhandled exceptions and returns a client return code 13 for errors in the client OCX and EXE programs. Location: System.ini

Miscellaneous INI Parameters

The following miscellaneous INI parameters are available for use within FormWare:

Parameter	Syntax	Description
CLEANUP	CLEANUP={D | B | I | O | E | T | M | S}	Cleanup images. Performs custom cleanup operations separate from those specified in the job. Setting this parameter overrides the cleanup options for the job. Both the CLEANUP parameter and the O and E parameter values must be in uppercase. Options: D - Clean up batch data files (BCH files)

Parameter	Syntax	Description
		B - Clean up BDF files
		I - Clean up image files
		O - Clean up OCR files
		E - Clean up enhanced image files
		T - Clean up thumbnail image files
		M - Clean up multi-page image files
		S - Clean up subdirectories; applies to images, enhanced images, and thumbnail images.
		You can specify one or more options, where each character translates to a different cleanup option.
		Location: System.ini, <User>.ini, <Group>.ini, <Job>.ini, environment variable SET CLEANUP=<Client Parameters of CLNCLNT.EXE JobFlow task>
CNO	CNO={Y | N}	Correct non-OCR fields. Flags a field with a field correction character for fields with no OCR results. Otherwise, the default data is written into the field (with no OCR results) and it is not stopped on during Completion.
		This is also a batch build command line parameter.
DOSTATS	DOSTATS={Y | N}	Generate recognition statistics. Provides the ability to generate recognition statistics reports from answer files.
		Location: Recognition engine INI file
EDITOR	EDITOR=<program>	Text editor. Specifies a text editor other than the FormWare text editor, such as Notepad.exe.
EUR	EUR={Y | N}	European date. Uses the European date format DD-MM-YY rather than MM-DD-YY.
FAXCVTDPI	FAXCVTDPI= <output DPI>	Fax conversion DPI. Enables 200DPI color or fax images to be converted to 300 DPI bi-tonal images in IPP.
		Specifies the DPI of a Faxed image converted to a Group4 TIFF using the IPP module. Use this parameter when selecting the Convert Image DPI (Fax) option on the Image Pre-Processing Options dialog for the job. The output DPI is the DPI of the converted image.
		Default: 200
		Location: System.ini, <Job>.ini, IPPClnt.ini
FEX	FEX={Y | N}	100+ data forms. Enables up to 16,000 forms per job. All form identification information is then expressed in 5 digits.
		Location: System.ini, <Job>.ini
FILES	FILES=<path>	Alternate files directory. Specifies an alternate FILES directory indicating that the BDFs are located in a directory other than \FormWare\Files, and that the batch data files should be created in the alternate directory rather than the default.
		Using the JobFlow Client eliminates the need for this parameter.
		This directory must already exist, and the parameter must include the full path to the directory, including the drive name.
		Example:
		FILES=C:\Work\Files

Parameter	Syntax	Description
FMP_FDP	FMP_FDP={Y | N}	Form/Field properties. Enables use of FW.SetProperty functions in a VBA project running during pre-Completion processes such as Recognition and Form ID.
		Location: System.ini, <Job>.ini
FORCE_UNKNOWN	FORCE_UNKNOWN= <form name>	Override unknown form types. Enables all unknown form types to be forced to a specific form number. This permits limited intermixing of free forms and fixed forms, by forcing unknown form types to a single free form type.
		Location: System.ini, <Job>.ini, FormID.ini, command line argument calling the Form ID component
FPOFILEEXT	FPOFILEEXT= <extension>	FPO File Extension. Defines the extension of FPO files.
		Location: System.ini, <Job>.ini
GROUPS	GROUPS= <group name>	Group. Specifies the group name to which the operator belongs.
ICAPTURE_ RECV_DELAY	ICAPTURE_RECV_ DELAY= <# milliseconds>	ICapture WSARecv Delay. This parameter is a work around for a problem in the Microsoft socket library to FWNetSrc.dll. This setting is not recommended and should be used as a last resort.
		Setting the value to 1 introduces a delay before WSARecv() to avoid a potential race condition.
		If the ICAPTURE_RECV_DELAY entry does not exist or is set to a value less than one, no delay is introduced.
		There is no maximum delay enforced, so you must take care.
		Location: System.ini, <Job>.ini
IMAP	IMAP=<inputmap>	Input map. The system overloads the input map with the <inputmap> found in the \FormWare\Keymap directory. You must create this file in a text editor and save it with an INI extension in the \FormWare\Keymap directory.
IPPCLNT_OUTPUT_ STROPTS	IPPCLNT_OUTPUT_ STROPTS={-t-N | SET_PHOTO=0 | SET_ PHOTO=1}	Defines the output image format in IPP.
		Options:
		t-N - Changes stripped Images to non-stripped images. When you include this parameter in the <Job>.ini file, it changes stripped Group4 images to non-stripped Group4 images. FormWare requires non-stripped images for processing. Therefore, if your scanner produces stripped images, you must add this parameter to your <Job>.ini file to change the images so FormWare can process them. This value is case sensitive and requires a lowercase t and an uppercase N.
		SET_PHOTO=0 - Changes white-on-black to black-on white. When you include this parameter in the <Job>.ini file, it changes white-on-black images to black-on-white images. If your images are already black-on-white, this parameter has no effect.
		SET_PHOTO=1 - Changes black-on-white to white-on-black. When you include this parameter in the <Job>.ini file, it changes black-on-white images to white-on-black images. If your images are already white-on-black, this parameter has no effect.
		Location: System.ini, <Job>.ini, <JobFlow>.ini, IPPClnt.ini

Parameter	Syntax	Description
LANGUAGE	LANGUAGE= <language name>	Foreign language support. Enables foreign language support for the AEG and CGK recognition engines. The <language name> value is the appropriate foreign language. Some valid values are: ENGLISH_US, GERMAN, FRENCH, DUTCH, or ITALIAN. Other languages are available. Location: <Job>.ini
LFN	LFN={Y | N}	Long filename support. Activates long filename support in the FormWare system. With this parameter set to Y, file names can be up to 27 characters long. Location: System.ini
LOGPATH	LOGPATH= <path to a file>	Defines where log information will be saved. Determines how FormWare components log error messages and other debugging information. You can add the LOGPATH parameter to an INI file to determine how information is logged to a disk file. The value must be a fully qualified path and file name. If you do not set the LOGPATH parameter or set it to empty (i.e., LOGPATH=), then no file logging is performed. Log files are always appended to, so it is important to delete or archive them periodically so that disk space is not compromised. You can use the following setting to dump the contents of the error stack whenever an error condition occurs: FWDEBUG=UF LOGPATH=<path to a file> Whenever a batch is processed that results in an error, the base client forces a "dump" of the entire error stack at the end of the batch. If no error occurs, then a dump is not generated. An error, in this case, is any return code from 2-99 excluding 11. Custom return codes are not included. Used by itself, the "U" value does not cause much to happen; however, you can add the "F" value to log to a file and the "V" value to dump all verbose messages that accrue while a batch is being processed to the file. Note: Be careful with the LOGPATH property so that separate processes or separate machines do not simultaneously write to the same file.
NORULES	NORULES={Y | N}	Disable Rules. No rules run during recognition. Disables the application of rules for the specified recognition engine. Location: <Recognition>.ini
OCRDATA	OCRDATA= <full path>	OCR directory. Specifies an alternate recognition results file directory. The system assumes that the image and OCR data files always reside in the same location. However, if you want them to reside in separate directories, use this parameter.
PERFECTPOST	PERFECTPOST= {Y | N}	PerfectPost. Causes the InScript recognition engine to start the PerfectPost program on the same machine running InScript. Location: <Job>.ini

Parameter	Syntax	Description
RMT_USERNAME	RMT_USERNAME= <user name>	Remote user name. RmtProc automatically does a Remote Sync in the case that a new or different user logs in. If the INI file contains a value for RMT_USERNAME and the user logs in using a difference name, an automatic Remote Sync occurs and the INI file is updated to reflect the new user name. Location: RmtXfer.ini
RMT_WORKONLINE	RMT_WORKONLINE= {Y	N}	Work online remote. When distributed completion is run with the RMT_WORKONLINE=Y option, and a operator interrupts a batch while the background download thread is still downloading batches, the remaining batches are left in an error state, and the server side BDF gets deleted.
STATS	STATS=<Path>	Statistics directory. Specifies an alternate folder for the operator statistics file. You can use this parameter if you want the statistics file created in a secure or specific directory; or if you already have a program that runs statistics and it expects to find the statistics file in a certain location. Default: \FormWare\Stats\
VBAMAXLEVEL	VBAMAXLEVEL= {CLIENT	TRANSACTION	DOCUMENT	IMAGE	FORM	FIELD	VBA maximum level. The VBA component makes event calls up to and including the specified level. If lower level events are not used, processing speed increases. Example: VBAMAXLEVEL=FORM The field level events are not called.
VBASETFOCUS	VBASETFOCUS= {Y	N}	Set VBA focus. Enables you to display data forms in the foreground that are displayed within VBA code. By turning this parameter off, the VBA component runs faster due to less display overhead.
YEAR	YEAR= {<4DIGIT>	<2DIGIT>}	Y2K. When you set this parameter to the 4 digit value, dates in the operator statistics file are recorded with four digits instead of two. Default: 2 digits (i.e., 00 instead of 2000) Location: System.ini

Client Parameters

Many FormWare components have client parameters or additional switches that cause the component to behave in a specific manner. Set these parameters in the **Client Parameters** field on the JobFlow Task Properties, General tab. The following tables list the client parameters available for each of the components allowed in the **Client Program Name** field.

Client parameters are grouped into the following categories:

- Completion Client Parameters
- Batch Build Client Parameters
- FPL and Export Client Parameters
- Miscellaneous Client Parameters

Completion Client Parameters

The following client parameters are available for the Completion module (Complete.exe).

Modes of Completion

The following client parameters are related to Completion modes:

Option	Description
-c	Correct. Starts the Completion module in Correct mode.
-e	New. Starts the Completion module in New mode.
-f	Remove Flags. Starts the Completion module in Remove Flags mode.
-q	Quality Check. Starts the Completion module in Quality Check mode.
-r	Append. Starts the Completion module in Append mode.
-s	IQC/Rescan. Starts the Completion module in IQC/Rescan mode.
-SCR=5SIL1B80O	Substitution pairs. Performs substitution pairs processing. This mode presents only the characters specified in the string. This parameter must follow a mode of Completion parameter.
-u	Update. Starts the Completion module in Update mode.
-v	Verify. Starts the Completion module in Verify mode.
-w	Add New Images. Starts the Completion module in Add New Images mode.
-x	Read Only. Starts the Completion module in Read Only mode.
-y	Identify Images. Starts the Completion module in Identify Images mode.

Multi-Pass Applications

The following client parameters are related to multi-pass applications. If you use any of these parameters, they must follow a mode of Completion parameter.

Note: The Completion client parameters must be combined to call the exact mode of Completion. For example, if you want to call Completion in Verify mode for a Numeric Only multi-pass application, the client parameter would be: `complete -v -DSP=16`

Option	Description
-DSP=<integer>	Specified. Starts the Completion module for the Specified mode. Range:1-15
-DSP=16	Numeric Only. Starts the Completion module for Numeric Only fields.
-DSP=17	Alpha Only. Starts the Completion module for Alpha Only fields.
-DSP=18	Alpha Numeric Only. Starts the Completion module for Alphanumeric fields only.
-DSP=19	Non-Numeric Only. Starts the Completion module for non-numeric fields only.
-DSP=20	Update Only. Starts the Completion module for Update Only fields.
-DSP=21	Verify Only. Starts the Completion module for Verify Only fields.

Batch Build Client Parameters

The following client parameters are related to batch build.

Option	Description
-CNO=Y	Correct no OCR data. Performs correction when there is no OCR data.
-MFB=Y	Multiple images. Set this option if there are multiple records per image.
-RFN=99	Reject form 99. Places a reject character in form type 99 so the operator must stop.
-RNE=<value>	Not error code. Routes the batch with a "not error" exit code. Default: 0

FPL and Export Client Parameters

The following client parameters are related to FPL and export programs (FWEdtLng or FWExpLng).

Option	Description
[-EDT] -DLL:<DLL Name>.DLL	Post-Completion edit. Executes the specified Post-Completion DLL against batches.
-EDB<:program name,device>	Pre-Completion edit. Executes the specified stand-alone Pre-Completion edit against the recognition results file and the BDF. You must specify the program name for this parameter. The job specified in the BDF must exist at the time this task is executed.
-EDT<:program name,device>	File edit. Executes the specified stand-alone file edit against batches. You must specify the program name for this parameter.
-EXB:C,<device name>	Export, comma delimited. Exports all data in a comma delimited format, corresponding to the Comma Delimited export option. You must specify the device name for this parameter.
-EXB:F,<device name>	Export, by position. Executes an export according to the export positions set in Designer on a field-by-field basis. No other formatting is done. You must specify the device name for this parameter. The export position is on the Field Properties, Checkbox Edits tab in Designer. This parameter corresponds to the Forms export option in the Administration module.

Option	Description
-EXB:G<device>	Export, form defined. Exports data in a form-defined format. Data is output to a text file in a description=value format, with the following syntax: Image=<image path> form=<form name> fieldname=<field data> You can open this file in a text editor to determine a method of outputting the data into another source
-EXB:N,<device name>	Export, no formatting. Exports data from record 1, field 1 to the end of the batch, working sequentially through the batch without any formatting. You must specify the device for this parameter. This parameter corresponds to the Data Records export option in the Administration module.
-EXP<:program name,device>	Export. Executes the export program, other than the program and device specified in the job properties. If none is specified, the information is taken from the batch header or job properties.

Note: When you use the -EXB switch, none of the job properties related to export, such as pad characters, fixed length records, and blocked records will apply to this switch. To enable these options, you must write a custom export program that will be executed with the -EXP switch. The -SOE switches work with the EXP parameters (e.g., `-EXB<:program name, device> -SOE:R)`.

Option	Description
-SOE:R	Skip on export, rejected record. Does not allow export of rejected records.
-SOE:N	Skip on export, non-display files. Does not allow export of non-display files.
-SOE:B	Skip on export, both. Does not allow export of either rejected records or non-display files.

The following parameters impact the permanent merging of a processed image with an overlay image (a dithered image that appears slightly gray and *fuzzy* when viewed in Completion). If you need to retrieve the images later for any reason, you can merge the original with the overlay to provide a more useful view of the data.

When the two images are merged, the overlay image remains large because it is seen as many black lines interspersed with spaces, rather than a solid black line that can easily be compressed. The following parameters enable you to select the method that works best for merging the JobFlow.

Option	Description
-EXB:M,<device name>	Merge with dithering. Merges the processed image with the dithered overlay image, resulting in a larger image size. You must specify the device name for this parameter.
-EXB:m,<device name>	Merge without dithering. Merges the processed image with the undithered overlay image, resulting in a smaller image size. You must specify the device name for this parameter.

Note: Do not put a space after the comma between the program name and device. If you type the entry with spaces, the parameter will not function correctly.

Miscellaneous Client Parameters

The following client parameters apply either to multiple components or to a component not discussed in previous sections of this appendix.

Option	Description
\INTERACTIVE	JFBatch exit codes. Displays a message if a batch fails to be added to a JobFlow queue. Use this switch when calling the JFBatch program from the command line.
-j	Join utility. Calls the Join utility that merges individual TIFF files into a single multi-page TIFF, based on TIFF file naming conventions. Use this option in a custom JobFlow task by using TSJClnt in the Client Program Name field.
-s	Split utility. Calls the Split utility, used to split a multi-page TIFF into individual TIFF files. This parameter is used in a custom JobFlow task using TSJClnt in the Client Program Name field.
-h -r -s -c	Remote Transfer Utility (RmtXfer.exe). Options: h - Performs a Remote Sync. r - Runs the Remote Config File editor. s - Runs the RmtXfer.ini editor. c - Runs in Continuous file transfer mode; only used for Distributed Scanning.
-USE=BCH	Use batch file for Join utility. Uses the images in the batch data file to join individual TIFF files into a single multi-page TIFF file. Use this parameter in a custom JobFlow task using TSJClnt in the Client Program Name field and –j in the Client Parameters field.
-VBP=<project name>	Use VBA project. Define Visual Basic project. Calls the VBA project for all components except Completion. The magnitude of what this value applies to depends on where it is used. The Completion module runs a VBA project with the same name as the job name. For all other FormWare components, you must specify a VBA project when using this parameter. Location: <Job>.ini, <Module>.ini, <User>.ini, <Group>.ini, <Recognition> .ini

Form ID Parameters

This section describes the Form ID training parameters for Dynamic Form ID. This section contains information on each keyword supported by Dynamic Form ID. Each word includes usage information and the legal parameter limits.

Keyword descriptions include:

- BORDER
- BOTTOM
- CONFTHRESH
- DISABLE_ORIENTATION
- DROPOUT_THRESH
- FORM
- FORMNAME
- FW_COLOR_THRESH
- LEFT
- MAX_SHIFT
- METHOD
- RIGHT
- SCAN
- SCANCLASS
- SPREADTHRESH
- TOP

BORDER

Usage

```
BORDER={
     }
```

Description

Used with the LEFT, RIGHT, TOP, or BOTTOM keywords, sets the border used as an "ignore" region during Form ID. Borders are useful during identification for ignoring portions of the form image that might throw off the identification process and cause Form ID to reject the image (such as cutting off edges or corners of the image, large black strips on the edge of the image due to bad scanning, etc.). Bordering also enhances detection of dropout versus non-dropout form images. Since the dropout versus non-dropout is computed EXCLUDING the border area, this can prevent shifted red (dropout) form images (such as those produced by the Kodak scanner) from being erroneously classified as non-dropout due to a black border.

Example

In the following example, a border of 0.25 inches on all sides of the form is ignored during Form ID.

```
BORDER={
   LEFT=0.25F
   RIGHT=0.25F
   TOP=0.25F
   BOTTOM-0.25F
}
```

Default

```
BORDER = (0,0,0,0)
```

BOTTOM

Usage

```
BOTTOM = <val>
```

Description

Ignore <val> inches of border along the bottom edge of the form image. In the following examples, the bottom 0.20 inch of the form image is ignored.

Example

```
BOTTOM = 0.20F
```

Range

The number specified must be less than or equal to the image height.

CONFTHRESH

Usage

```
CONFTHRESH = <val>
```

Description

Sets the confidence threshold that determines the "closeness" of match between the run-time image and the training image. The confidence must be greater than CONFTHRESH to accept the image.

Example

```
CONFTHRESH = 0.85F
```

Range

The number specified must be in the range 0.0F to 1.0F, which corresponds to the confidence range of 0 to 100 percent.

DISABLE_ORIENTATION

Usage

```
DISABLE_ORIENTATION = <val>
```

Description

If this keyword is set to 1, orientation detection is disabled. In this case, Form ID does not train the three non-zero orientations (90, 180, and 270 degrees) and does not attempt to detect rotation at run-time. This is useful for increasing acceptance and accuracy on images that are scanned in the same direction. It also increases throughput performance at run-time. If FormID is processing color images this option must be disabled. Specify this keyword in the <Job>.trn file.

Example

```
DISABLE_ORIENTATION = 1
```

Range

This parameter may be set to 0 or 1.

Default

The default value is 0 or detection enabled.

DROPOUT_THRESH

Usage

```
DROPOUT_THRESH = <count>
```

Description

Sets the blackness threshold used for discriminating between non-dropout and dropout images. If less than <count> number of pixels are "black," the image is classified as a dropout form and assigned to a dropout form internal scan class; otherwise, it is assigned to a non-dropout scan class. <count> is computed from a down sampled 64 x 64 version of the original image.

Example

```
DROPOUT_THRESH = 410
```

Range

The range for this word is any positive integer between 0 and 4096.

Default

The default for this word is 430.

FORM

Usage

```
FORM={
}
```

Description

Allows you to set parameters specific to a particular data form type. Must be used with the keyword FORMNAME.

Example

```
FORM = {
    FORMNAME = MY_FORM_1
    CONFTHRESH = 0.80F
}
```

FORMNAME

Usage

```
FORMNAME = <FDF name>
```

Description

Sets the data form type to the name of the Form Definition File (FDF). The FDF name is 8 characters in length and must be uppercase.

Example

```
FORMNAME = MY_FORM_1
```

FW_COLOR_THRESH

Usage

```
FW_COLOR_THRESH = <number 0-255>
```

Description

Color Threshold. Adjusts the sensitivity of the Color Form ID operation. Larger values allow for a range of brightness away from the exact value for successful form ID. Set this parameter in the FIDClnt.ini file.

Example

```
FW_COLOR_THRESH = 150
```

Range

The legal range for this parameter is 0-255 (units are in brightness).

Default

The default for FW_COLOR_THRESH is 10.

LEFT

Usage

```
LEFT = <val>
```

Description

Ignores the specified number of inches along the left edge of the image. In the example below, the left-most 0.20 inches of the image is ignored.

Example

```
LEFT = 0.20F
```

Range

The specified number of inches must be less than or equal to the image width. If larger, an error message is reported during training.

MAX_SHIFT

Usage

```
MAX_SHIFT = <shift>
```

Description

Specifies the maximum distance (in inches) that an image is shifted both horizontally and vertically at run-time prior to matching the run-time image with the training image. This "dynamic shift correction" increases Form ID acceptance and accuracy on images whose x or y position is unreliable due to bad scanning. It is applied in the range of + or - <shift> inches. The larger the value <shift>, the more processing time used. A value equal to 0.25F is recommended. Set this value to 0.0F for maximum processing speed if throughput is an issue.

Example

```
MAX_SHIFT = 0.15F
```

Range

The value specified must be less than or equal to the image size (in inches).

Default

The default for MAX_SHIFT is 0.25F.

METHOD

Usage

```
METHOD = <method>
```

Description

Specifies whether to run the quadrant method of Dynamic Form ID. The quadrant method is used to automatically "zoom" into small areas of the image to discriminate subtle differences between form types. METHOD is used to enable the quadrant method within a single form type (FORM), a single scan class (SCAN), or globally (FIDINFO).

Example

```
METHOD = QUADRANT
```

Range

Two choices are available: QUADRANT and NORMAL.

Default

The NORMAL method is the default.

RIGHT

Usage

```
RIGHT = <val>
```

Description

Ignores the specified number of inches along the right edge of the image. In the example below, the right-most 0.20 inches of the image are ignored.

Example

```
RIGHT = 0.20F
```

Range

The specified number of inches must be less than or equal to the image width. If larger, an error message is reported during training.

SCAN

Usage
```
SCAN={
}
```

Description

Allows you to set parameters specific to a scan class. May be used with the scan class naming keyword SCANCLASS.

Example

```
SCAN = {
      SCANCLASS =MY_CLASS1
      CONFTHRESH = 0.80F
}
```

SCANCLASS

Usage

```
SCANCLASS = <name>
```

Description

Sets the "name" tag for a user-selected scan class.

Example

```
SCANCLASS = MY_CLASS_1
```

Range

The tag is a character string of less than 64 characters.

SPREADTHRESH

Usage

```
SPREADTHRESH = <val>
```

Description

Sets the spread threshold that defines how closely the Form ID best result and the runner up result match. The difference in confidence between the first and second guess must be greater than <val> amount for the best guess to be accepted.

Example

```
SPREADTHRESH = 0.15F
```

Range

The number specified must be in the range 0.0F to 1.0F, which corresponds to a spread range of 0 to 100 percent.

TOP

Usage

```
TOP = <val>
```

Description

Ignores the specified number of inches along the top edge of the image. In the example below, the topmost 0.20 inches of the image are ignored.

Example

```
TOP = 0.20F
```

Range

The number specified must be less than or equal to the image height. If larger, an error message is reported during training.

9 Glossary

A

Acceptance Test Plan (ATP): An ATP is comprised of test cases and scripts prepared to facilitate the testing process. The acceptance criteria are determined by the requirements in the Detail Specification.

Accumulator: An accumulator is one of 16 numeric data storage areas. You can add or subtract field data to the accumulator value. The purpose of accumulators is to achieve a balance value of zero at the end of the batch.

Alpha: Alpha or Alphabetic refers to uppercase and lowercase characters in the 26-character alphabet and the space characters.

Alphanumeric: Alphanumeric refers to all printable characters in the ASCII code set, including letters, numerals, and symbols.

Apache: Apache is a public domain Web server used under many platforms. We support Apache under Linux.

Application Programming Interface (API): The Application Programming Interface is a program language used to create DLLs.

Auto Functions: Auto functions allow you to perform specific types of functions on fields within a record or a data form. The auto functions are set for the field on the Selective Edits tab of the Field Properties window. You use auto functions during the completion process by turning the [F2] auto function key OFF or ON.

Auto Length Detection (ALD): Auto Length Detection is a feature supported on certain scanners that allows you to scan documents of different sizes. A scanner with the ALD feature automatically detects the size of the document you are scanning. Scanners without the ALD feature pad each image out to the default page size.

AutoFind: AutoFind is a data removal feature that can locate and remove horizontal and vertical lines you designate for removal when you set up the data form.

Automated Data Edit (ADE): An Automated Data Edit contains highly formatted data. This includes, but is not limited to, data such as phone numbers, dates, and addresses. Within Designer, the Automated Data Edits feature enables you to create fields that will contain formatted data and further define the type of data to be contained within those fields. You can also apply the automated data edits during the Recognition and Completion modules.

Automatic Document Feeder (ADF): An Automatic Document Feeder lets you feed multiple paper documents or forms into a scanner and continuously scan those documents to create image files.

Aux Dup Table: The Aux Dup table allows you to define insertion values in a table that Completion operators can insert into fields during the data entry process. You can define and use only one Aux Dup table per job. For example, using the Designer module, you can create an Aux Dup table that contains commonly used terms such as "Street," "Avenue," "Boulevard," etc. Then, during a Completion session, an operator can press a shortcut key (i.e. the [Home] key), retrieve this list, and select a value to insert into the current field.

B

Bar Code: A bar code is a deliberate pattern of vertical stripes and spaces used to identify an item. Within a forms processing application, you can use bar codes to identify form types or extract data.

Base File Directory: The Base File Directory is the directory path to where Batch Definition Files are created using the FormWare Scan module. The Base File Directory path is set on the Batch tab of the File Management Properties window in the Scan module. The default path is \FormWare\Files. For example, if you set the Base File Name to "BAT00000.bdf," all subsequent BDF files will follow this naming convention. The Base File Directory path cannot be the same as the Base Image Directory path.

Base File Name: The Base File Name sets the base naming convention for the Batch Definition Files created using the FormWare Scan module. The Base File Name is set on the Batch Property Page of the File Management Properties window in the Scan module. The values set in the Update Properties editor on the Batch Property Page determine what portion of the filename to increment for each new BDF file.

Base Image Directory: The Base Image Directory is the directory path to the scanned or imported image files (.tif and .jpg files) created using the FormWare Scan module. You set the Base Image Directory path on the Image tab of the File Management Properties window in the Scan module. The default path is \FormWare\Images. The Base Image Directory path cannot be the same as the Base File Directory path.

Base Image Name: The Base Image Name sets the base naming convention for the image files. The .tif extension is the default for image files but is not required. The Base Image Name is set on the Image Property tab of the File Management Properties window in the Scan module. For example, if you set the Base Image Name to "HCFA0000.tif," all subsequent image files will follow this naming convention. The values you set determine what portion of the filename to increment for each new .tif file.

Batch: A batch is a collection of paper documents that you want to process together. A batch can be grouped on any user-defined criteria—a specific number limit (e.g., always 50), a geographical location, or a numeric value—anything that is logical to your organization. The term "batch" is commonly used to refer to this group of documents as they proceed through the forms processing application.

Batch-Based Reject Repair: Defined in the Designer module, batch-based reject repair allows Completion operators to view all rejected characters for the batch in the reject ticker tape instead of just the rejects for the current record.

Batch Data File: A batch data file contains the data, or information, captured from a batch of paper forms—either by recognition technology, manual data entry, or a combination of the two. The batch data file is a binary file that only FormWare modules and programs can modify.

Batch Definition File (BDF): The Batch Definition File is a text file that contains the job name, a list of images associated with a batch, and the order those images are to be processed. You can identify BDF files by the .bdf extension (i.e., <FileName>.bdf).

Batch Log File: A batch log file is a text file that contains error or audit log information. Output statements from field edit or file end edit programs generate the batch log. You can identify batch log files by their .log extension.

Batch Mode: Within the Scan module, the batch mode indicates at what point in the scanning process to create a new Batch Definition file. There are three batch modes: Count, Ticket, and Separator, with a Subdirectory option.

C

Cascading Style Sheets (CSS): Cascading style sheets are a feature of HTML. A style sheet is a template describing how things should be displayed within a browser. Style sheets are referred to as cascading because more than one can be applied to each web page. CSS are not fully supported by any Web browser.

Character Smoothing: During Image Pre-Processing, character smoothing evens out the edges of characters on an image (removing small bumps or pits) to improve recognition results.

Check Digit: A check digit is the final digit of a numeric field that is generated by executing an algorithm on each previous digit in the field. Check digits provide security and validation for highly sensitive numeric data. The algorithms are based on a field property and can be used to validate or generate a check digit.

Cleanup: The Cleanup function removes all files associated with batches listed in the Cleanup JobFlow task. This includes deleting all image files, recognition results files, batch data files, and BDF files associated with the batch.

Client Return Codes: Client return codes are generated when a batch is processed or fails processing. These client codes return information on the processing state or reason for failure of batches within the JobFlow system.

Common Gateway Interface (CGI): CGI is a mechanism through which web servers handle dynamic files. When a request is made to a file located in the CGI-BIN directory, rather than reading and sending the page, the file is executed and the output is sent rather than the page directly.

Conditional Link: The conditional link function is a field property that, when set up for a field on a data form, allows operators to type a specific form number to jump to that data form in the job.

Confidence Level: The confidence level is a numeric value between 0 and 100 that the recognition engine returns for each character it recognizes in a batch. This number is a percentage of how confident the engine is about the character it returned. Characters that do not meet the confidence level set for specific fields in the Designer module display in the reject ticker tape during the Completion process.

Confidence Threshold: During form identification, the confidence threshold is the level that must be met to register the data form. When the recognition engine cannot find registration anchors, it can be an indication that this confidence threshold is set too high.

Contrast: The contrast is the range between the lightest and darkest tones on a paper document. The default contrast is 50%. During the scanning process, adjusting the contrast can improve recognition of the image by removing unwanted speckles or combs. The contrast is set for images in the Scan module on the Scanning Properties window, Current Settings tab.

Cropping: During Image Pre-Processing, image cropping refers to the process of removing unnecessary edge material from the image. Cropping an image reduces the image size and can save on hard disk space.

D

Data Form: The data form is a collection of fields that defines the data you want to extract from an image or paper form. You create the data form using the Designer module. The data form displays on the lower portion of the screen to the operator during a Completion session. The operator then inputs data into the fields on the data form.

Data Presence: Data presence is a method of identifying a data form. This option is helpful when you use an item such as a company logo for form identification. To make this association, a field is defined on the data form in the Designer module.

Despeck: During Image Pre-Processing, despeck removes isolated black specks from images. This reduces the amount of "noise" on an image and can improve recognition accuracy.

Destination Directory: In the Scan module, the destination directory is the directory path where imported image files are stored. This directory path is set in the Base Image Directory field on the Image tab of the File Management Properties window within the Scan module.

Diagonal Protection: Entering a negative value for Isolated Despeck activates Diagonal Protection. The speck removal process then looks at the four corners in addition to above, below and either side for pixels close to the speck targeted for removal. Use this feature if despeck is removing diagonal portions of characters.

Display Mode: The Display Mode is a field option used for multi-pass forms processing applications. Each numeric Display Mode value represents a different Completion pass. The fields and options defined for each Display Mode pass are completely user-defined.

Distributed Scan: Distributed scanning enables you to scan documents on remote scanners and import images from remote files. Rather than using local scanning processes when your server and your scanner hardware reside in one physical location, you can use distributed scanning processes when your server and your scanner hardware reside in separate physical locations.

Document: A document is a multi-page form grouped together for forms processing purposes. For example, a tax form and its related schedules is a document. Within each document, there is a primary form called the document indicator form, which is defined by the Auto ID Linked Forms option on the Form Properties, ID tab in the Designer module. Each time the Completion module encounters a form of this type within a batch, it increments the document count by one.

Dot Shading: Dot shading refers to dithering that creates shades of gray on a paper form. On paper forms, shaded areas emphasize features on the form. However, on an electronic image these shaded areas take up unnecessary disk space and make recognition difficult. During Image Pre-Processing, the dot shading feature removes these shaded areas on the image. This reduces image file size and produces a cleaner, clearer image that improves recognition processing.

Dot Verification: Dot verification, a related function of dot shading, allows you to refine the dot shading removal process.

Dots Per Inch (DPI): In the Scan module, Dots Per Inch refers to the number of dots per inch, or resolution, of the scanned image. DPI also determines the size of the image file. Large image files require more processing time and take up more disk space. Therefore, you should set the image DPI at the lowest value that still produces a clean, readable image.

Dynamic Form ID: Dynamic form identification trains a set of assigned form images to identify the production images.

Dynamic HTML (DHTML): Dynamic HTML is a method of generating HTML pages/files dynamically upon request, rather than accessing static HTML files.

Dynamic Link Library (DLL): Within the FormWare system, a Dynamic Link Library is an edit program that contains special FormWare function calls. You can create one or more DLLs to use during the Completion process. These DLLs contain FormWare function calls that are unique to your forms processing application.

E

Edit Server: An edit server is a workstation attached to a network that polls the JobFlow queue files, looking for batches to process. When it finds a batch, the server runs one or more edit programs against this batch. Edit servers can launch one or more edit programs against batches without human interaction.

Error Flag: Used during the Completion process, operators can insert error flags into designated fields or records to identify those places requiring correction. With non-destructive reject repair, the default behavior, a field or record is flagged when it is rejected for later correction while maintaining the field data. With destructive reject repair, a character is inserted into the data field(s) to signal an error for later correction.

Errors Allowed: The number of low confidence characters allowed in a field before the entire field is rejected.

Export Program: An export program is a program written in the FormWare fourth-generation programming language (FPL) that is run against a batch or a group of batches. The export program formats the batch data for printing or export to a text file.

Extensible Markup Language (XML): Extensible Markup Language is a markup language for documents containing structured information, including text and graphic content, and the constraints or rules that indicate what role the content plays. Similar to HTML, XML uses tags (text bracketed by < >) and elements (data between tags). Unlike HTML, XML does not specify what each tag means or how the text between tags will appear in a web browser.

F

Feature Form ID: Feature form identification uses elements of the setup image specified in the Designer module to identify the form type.

Field Edit: A field edit is a FormWare program executed during a Completion session to examine, verify, and/or manipulate the data being entered in the field.

Field Type: The field type refers to the type of characters accepted within the field on your form. Valid field types are: Any Type, which accepts any type of characters; Alpha Only, which accepts only alphabetic characters; Uppercase, which accepts only alphabetic characters in uppercase; Unsigned Numeric, which accepts only unsigned numeric characters; Signed Numeric, which accepts signed or unsigned numeric characters; and Mostly Numeric, which accepts mostly numeric characters.

File Edit (File End Edit): A file edit is a FormWare program that is normally executed after the Completion session is terminated. Additionally, a file edit program can be used independently for applications programming. File End Edits are always executed on termination of a batch from within Completion. Standalone file edits can be executed at any time during the JobFlow process.

Form Definition File (FDF): A job contains one or more data forms, or Form Definition files. Data forms contain the fields where data is inserted during data completion. By creating a data form, you create a form definition file. The FDF contains all of the display, recognition, and completion information assigned to the related data form. This file is saved in the \FormWare\Forms folder.

Form Identification (Form ID): Form identification is the process of relating production images with a data form. There are two basic types of form identification within FormWare: manual and automatic. Automatic Form ID is a set of processes the FormWare software performs prior to a Completion session, while manual Form ID is a process an operator performs during a Completion session. There are three different methods of Automatic Form ID: Dynamic, Feature, and Implicit.

Form Removal: Form removal is the process of removing extraneous data from an image.

Free Form Recognition: Free form recognition is technology for locating and extracting data from structured or unstructured documents in which the location of the data may vary from image to image. At the field level, free form fields are defined, including the necessary pattern-based information (keyword context and data pattern) for determining how to locate and extract the data.

H

Hypertext Transfer Protocol: HTTP is the acronym for Hypertext Transfer Protocol. The client/server protocol used to access information on the World Wide Web.

I

I-Form: I-Forms are forms that can be accessed over the Internet within a browser. This requires that you set Internet-specific parameters at the job level. These parameters control the completed and canceled transactions associated with the I-Form you are creating, as well as defining scripting options and JobFlow setup routines.

Image: An image is a digital representation of a paper form or document. All image files within the FormWare system must be stored as Group 4 TIFF files.

Image Deskew: The image deskew property straightens the image on the page. The Deskew feature is accessed from the Image Pre-Processing page and provides options to set the degree of horizontal or vertical skew acceptable.

Image Pre-Processing (IPP): Image Pre-Processing cleans up your images to make them "recognition ready." A necessary, but often overlooked part of any forms processing system, IPP can eliminate many of the problems associated with forms processing applications that use recognition technologies. Pre-processing images can improve readability and, ultimately, the accuracy rate of your images.

Image Processing : Image Processing performs image manipulation and enhancement, both important operations when using recognition technologies. The Image Processing module consists of five distinct processes, each of which is crucial to the success of your forms processing application. These processes are Image Pre-Processing, Image Quality Check, Form Identification, Image Enhancement, and Recognition.

Image Quality Check (IQC) : Image Quality Check refers to the verification and correction of the quality of the images. There are two types of Image Quality Check processes: Automatic IQC and Manual IQC.

Image Zone: An image zone is an area on an image that relates to a particular field on a paper form or document. An image zone differs from a recognition zone in that it does not contain special properties used during the recognition process. You can use image zones in Key from Image (KFI) applications.

Implicit Form ID: Implicit form identification applies either feature or dynamic identification to the first image in a document. The remaining images are assigned form identification numbers without applying identification technology.

In-Context Viewport: The in-context viewport is the center section of the post-recognition completion window that displays the image area containing questionable characters and low confidence characters. The viewport shows zone highlighting in one color and low confidence characters in a contrasting color.

Index: An index is a file created to allow direct access to records in a data file. Each index entry consists of a "key," which is composed of information pertaining to the referenced record, and a link to the record in the data file.

INI File: INI parameters are set in INI files, which are text files that control the various functions of the FormWare modules and processes. Each parameter in an INI file must be placed on a line by itself.

Intelligent Character Recognition (ICR): Intelligent Character Recognition (ICR) is an advanced form of object code recognition technology used for hand print data.

Intensity: Intensity is the strength of color black against white.

Internet FormWare Server (IFS): Internet FormWare Server is a FormWare program used for I-Form validation, transaction management, and data translation into JobFlow. The IFS provides for a highly scalable solution and can perform editing functions with VBScripts developed within Designer.

Internet Information Server (IIS): IIS is Microsoft's brand of Web server software, using Hypertext Transfer Protocol to deliver World Wide Web documents. It incorporates various functions for security, allows for CGI programs, and provides for Gopher and FTP servers.

Inverse Text: Inverse text is white text on a black background.

Isolated Despeck Height or Width: This is activated in Isolated Despeck Width when a negative value is entered in the field. This algorithm takes the values in Isolated Despeck and Isolated Despeck Width and constructs two virtual rectangles. Speck Removal removes any black specks that fit within that area. For example Isolated Despeck value = 2 and Isolated Despeck Width value = -3 produces a 2x3 pixel rectangle and a 3x2 pixel rectangle.

J

Job: A job is a collection of data forms. Each job can point to as many as 99 different forms or Form Definition files. A job contains the properties and settings that affect how the forms associated with the job are processed.

Job Definition File (JDF) : A Job Definition File is a binary text file that contains basic information about the job and how to process the forms associated with the job.

JobFlow: The JobFlow module is specialized workflow software for forms processing. JobFlow routes batches through image pre-processing, to recognition engines, to the Completion module for data entry and verification, and finally to the Export step.

K

Key: A key is an index field that consists of the characters of data being entered.

Key-from-Image (KFI): In a Key-from-Image application, the operator manually identifies images (keys in the correct form number) during Form Identification and types data into the data form portion from the Image Viewport during Completion. You can design a KFI application by selecting the Image Only option on the Form Properties dialog, Image property.

Key-from-Paper (KFP): In a Key-from-Paper application, the operator manually types information into the data form from the hard copy document rather than from an electronic image.

Keyword: A keyword is a feature of Free Form Recognition, defined in Designer (Field Properties/Recognition page) as a field searchable by a recognition engine (currently InScript, CGK, or AEG). By locating the keyword, Recognition can locate the data for that field.

L

Location Field: Within the Completion module, the location field is the farthest point in the record or batch that the cursor has ever been. For example, if the operator positions the cursor to Field 12 within a record and then returns to Field 1, Field 12 is the location field. The Completion module uses the location field to understand where the operator is within the record and the completion process. The program then knows when batches and records are complete and when to write record data out to data files. To access the location field within the Completion module, press [F7] (the location return key) or select Current from the Field menu on the toolbar.

M

Mark Sense Recognition: Mark Sense Recognition recognizes the checkmarks, Xs, or other marks placed in checkboxes. Relates to the industry term Optical Mark Recognition (OMR).

Masked Zone: Masked zones are image zones that are displayed one at a time to the operator during a Completion session.

Module Definition File (MDF): A Module Definition File (MDF) contains the variables used to export data from the FormWare software back to the InputAccel software.

Multi-Pass Application: A multi-pass forms processing application separates the batch-based reject repair process into more than one pass. Using the multi-pass feature, you can separate the reject repair by a specific field type or other criteria so that an operator sees only those fields that meet the specified type or criteria in the pass he is repairing. There are two multi-pass methods: field type and display mode.

O

Optical Character Recognition (OCR): Optical Character Recognition recognizes machine-printed alphanumeric characters that have consistent, predictable shapes. This technology converts documents that you can read into documents that your computer can read. During the conversion, the document is analyzed, and characters and words are saved as editable text.

Overlay Image: An overlay image is an image created with a blank paper form and scanned with a bulb that does not cause the form lines and boxes to drop out. This image is then dithered, or grayed, and combined with the production images to give the operator a guide or representation of the paper form.

P

Paper Document: A paper document is a collection of papers that you want to group together or have an existing relationship to one another. For example, a tax form with related schedules.

Paper Form: A paper form is a single-page paper document that you want to extract information from using forms processing technology (e.g., a single-page tax form).

Patch Code: A patch code is a feature supported by particular scanners and used to identify scanned images and to separate batches. During scanning or importing, a patch code event can also be set to trigger string segment (filename) updates.

PerfectPost: PerfectPost is a third party database that can be purchased as an add-on component to FormWare. The PerfectPost database contains known street, city, state, and zip code addresses within the United States.

Prepend: Prepend means to prefix or add to the beginning. It is the opposite of append.

Priority Number: JobFlow software allows batches to be assigned priority numbers, which allow batches to move to the front of the queue so priority batches are processed first.

Property Page: A property page displays the fields, check boxes, radio buttons, drop down lists, buttons, etc., for a specific FormWare program function. Settings on a property page impact the design of your forms processing application. You access a property page by selecting a tab on a dialog box.

Public Key Infrastructure (PKI): PKI is a system of digital certificates, Certificate Authorities (CA), and other registration authorities that verify and authenticate the validity of each party involved in an Internet transaction.

R

Recognition: Using one or more engines to convert the pixels of the text into characters that will be inserted into the fields of the data form.

Recognition Zone: A recognition zone is a type of zone or area on an image where you have assigned specific recognition properties. Among other things, the properties you set for the zone indicate the specific type of data you want to extract using recognition technology.

Record Auto Functions: Record auto functions allow you to perform specific functions on fields within a record when the auto function is turned on during the completion process.

Registration: During Image Pre-Processing, image registration refers to the settings that control consistent left and upper margins on the image. There are three different registration settings: Horizontal Registration, which maintains a consistent left margin for the image; Vertical Registration, which maintains a consistent upper margin for the image, and Registration Control, which fine-tunes vertical and horizontal registration settings.

Registration Anchor: A registration anchor is a mark placed on the data form that the recognition engine recognizes as a form registration mark. The registration anchor is a feature of the image, not the data form. A field on the data form can then reference that anchor (by creating a zone around it) to be used during registration.

Regular Expression: A regular expression in FormWare is a Free Form Recognition feature used to locate field values. Regular expressions use special characters (operators) and syntax that enable the program to return data from a searchable keyword field.

Reject Repair: Defined in the Designer module, reject repair allows Completion operators to view rejected characters for the current record.

Reject Ticker Tape: The reject ticker tape is the top portion of the post-Recognition Completion window that contains out-of-context, questionable, or low-confidence characters.

S

Scan Designer module: Scan Designer enables you to create or modify scan profiles, and settings needed to perform a scanning job and saved in a Scan Configuration (SCG) file.

Scan module: The Scan module includes three separate components that determine how images enter into the FormWare system. The scanning process is the input mechanism for scanning or importing electronic image files.

Scan Operator module: Scan Operator enables you to select SCG files, scan documents, create BDFs, import images, and add batches to a JobFlow.

Separator: During the scan process, a batch separator is a single piece of paper inserted at the beginning of and throughout a group of papers to be scanned. The batch separator must be a blank page that is the same size as the other pages. When the Scan module encounters the batch separator, it creates a new Batch Definition file with the image immediately following the batch separator as the first image in the batch.

Skew: Skew is a condition that occurs when a scanned document is not aligned horizontally and vertically.

Snippets: Snippets are portions of the image interspersed amongst the fields of the data form.

Submit Type: Submit Type options are set in Designer on the Field Properties, I-Forms page. The options refer to the type of transaction submission action associated with a button or bitmap. You can set the submit action for the button or bitmap to None, Previous, Cancel, Next, Done, Clear, Resume, or Suspend. When the user selects this button or bitmap on the I-Form, whatever submit type has been defined is activated.

T

Tab: A tab is the part of a dialog box you select to access a property page. One or more tabs can display on a single dialog box.

Thumbnail: A thumbnail is a function that allows you to view an entire image (in miniature) as it corresponds to the image viewport window. Thumbnails appear in other sections of the software—IPP, image viewer, and scanning.

Ticket: During the scan process, a batch ticket is a single piece of paper inserted at the beginning of and throughout a group of documents to be scanned. The batch ticket must be 20% larger or smaller than the other documents. When the Scan module encounters the batch ticket, it creates a new Batch Definition File (BDF) with the batch ticket image as the first image of the batch.

TIFF: TIFF is the acronym for Tagged Image File Format. It is a standard file format commonly used for scanning, storage, and interchange of gray-scale graphic images. FormWare supports TIFF files with the standard TIFF tags where the image compression used is Group IV 2D compression (non-stripped). Most scanners produce this format.

U

Universal Resource Locator (URL) : A URL is an address on the Internet used by Web browsers to locate Internet resources. It specifies the protocol for accessing the resource (such as http: for a World Wide Web page or ftp: for an FTP site), the server on which the resource resides (such as //www.whitehouse.gov), and, optionally, the path to a resource (such as an HTML document or a file on that server). The URL is used for I-Forms to locate the web page where the user is taken to once a transaction has been submitted.

User-defined Field (UDF) : User-defined fields are common data fields—such as age, date, address, city, and zip code—which are found on many different forms. You can define these fields once with associated field properties, and then access them for different jobs.

V

Value Table: A value table is a data file or index used in completion or edit programs such as a lookup table to perform replacement, validation or display functions. For example, a value table can be used to expand a zip code into city and state information. Value tables cannot be used in edit programs.

VBScript: Microsoft Visual Basic Scripting (VBScript) is a fast, portable, lightweight interpreter for use in World Wide Web browsers and other applications that use ActiveX Controls, Automation servers, and Java applets.

Verification: During the verification process, each character returned by the recognition engine or reject repair operator is compared to the character's type by the verification operator. The verify operator does not see the data entered by the recognition engine or the repair process. If a discrepancy is encountered, the verify operator will be presented a message. An operator can change or accept the original text depending on the cause of the error.

Visual Basic for Applications (VBA): Visual Basic for Applications is a programming language that is accessible from within FormWare modules, such as Designer. VBA allows you to design special programs that will customize your application.

W

Web Server: The Web Processing Server is a FormWare component that must be installed on the Web server machine. Internet Information Services (IIS) and Apache are both supported as the web server software.