IBM FileNet Capture and IBM Datacap

Kevin Bowe

Social

Information Management

Introduction

This IBM® Redpaper™ publication has various objectives. It uses a fictional capture processing scenario to identify the differences between IBM FileNet® Capture and IBM Datacap. Using native capabilities, you see how each product processes the same form. Additionally, the paper reviews high-level attributes of an advanced capture solution and how they meet the scenario requirements. At the conclusion of this paper, you will have an understanding of how each product meets both the scenario requirements and supports the attributes of an advanced capture solution.

> **The intent of this paper:** This paper is not an application development guide. There are no tools to convert from one product to another. This paper explores only the native capabilities of the two products.

Content capture processing scenario

This paper references a single processing scenario (scenario). A fictitious financial institution, Bank B, is streamlining their capture processes. They are a global financial services company, with multiple headquarters, operation centers, satellite offices, and retail locations worldwide. Multiple offices are responsible for ensuring bank statements are processed and committed to the corporate IBM FileNet P8 Content Repository. Implementing process efficiencies means the bank must acquire and process bank statements by using a solution with the following capabilities:

► Ability to scan paper and index by using a web interface
► Ability to scan paper and index by using a desktop interface
► Ability to scan paper and index by using multifunctional devices
► Ability to acquire emailed forms
► Ability to acquire faxed forms
► Ability to acquire forms in an electronic format
► Automatically find and capture form data in varying locations
► Validate captured form data
► Route problematic forms to subject matter experts (SMEs) for resolution
► Manage all batches in the capture system throughout their lifecycle
► Provide access to real-time batch and user processing metrics
► Commit form and form data to any number of content repositories

Figure 1 shows a sample bank statement.

Sample Bank B

Sample Bank B N.A.
PO Box 2222
Bank City, CA 02222-0222

July 1, 2014 through July 31, 2014
Primary Account: **000001122334455**

CUSTOMER SERVICE INFORMATION

WebSite:	www.SampleBankB.com
Service Center:	1-800-123-1111
Hearing Impaired:	1-800-123-1112,
Para Espanol:	1-877-123-1113
International Calls:	1-001-123-1114

0001122 AAA 001 LA 0111- AAAT 1 00000000 01 0000
Bob Jones
Mary Jones
222 Mary Street
Bob City, ST, 11111

All references to names, places, address, and other data, are fictitious and are used for illustration purposes only

CHECKING SUMMARY

	INSTANCES	AMOUNT
Beginning Balance		**$61,807.40**
Deposits and Additions	14	888,904.32
Checks Paid	5	-6,345.28
Other Withdrawals, Fees & Charges	8	- 25,565.14
Ending Balance	**20**	**$918,801.30**

This message confirms that you have overdraft protection on your checking account.

TRANSACTION DETAIL

DATE	DESCRIPTION	AMOUNT	Balance
	Beginning Balance		$61,807.40
07/01	Deposit	9,213.51	71,020.91
07/09	Deposit	26,410.29	97,431.20
07/10	Deposit	11,424.00	108,855.20
07/10	Transfer from Saving	700,000.00	808,855.20
07/13	Withdraw	-2000.00	806,855.20
07/15	Deposit	9,213.51	816,068.71
07/15	Deposit	26,410.29	842,479.00
07/18	Deposit	4,120.17	846,599.17
07/18	Check Withdraw	-1,500.00	845,099.17
07/18	Check Withdraw	-410.21	844,688.96
07/19	Deposit	35,138.00	879,826.96
07/19	Withdraw	-1,323.21	878,503.84
07/20	Withdraw	-19,114.00	859,389.84
07/21	Check Withdraw	-390.63	858,999.21
07/22	Check Withdraw	-521.72	858,477.49

Continue to next page

Page 1 of 2

Figure 1   Sample bank statement

Both FileNet Capture and Datacap possess capture processing abilities.

IBM FileNet Capture Professional configuration

When creating a FileNet Capture application, you primarily focus on configuring the following four areas:

► Capture path
► Settings collection
► Template
► Index user interface

Capture path overview

A capture path identifies components and the order they are run. Components perform actions, such as scanning, image verification, assembling, indexing, and committing. FileNet Capture runs the capture path that passes a batch of work from component to component.

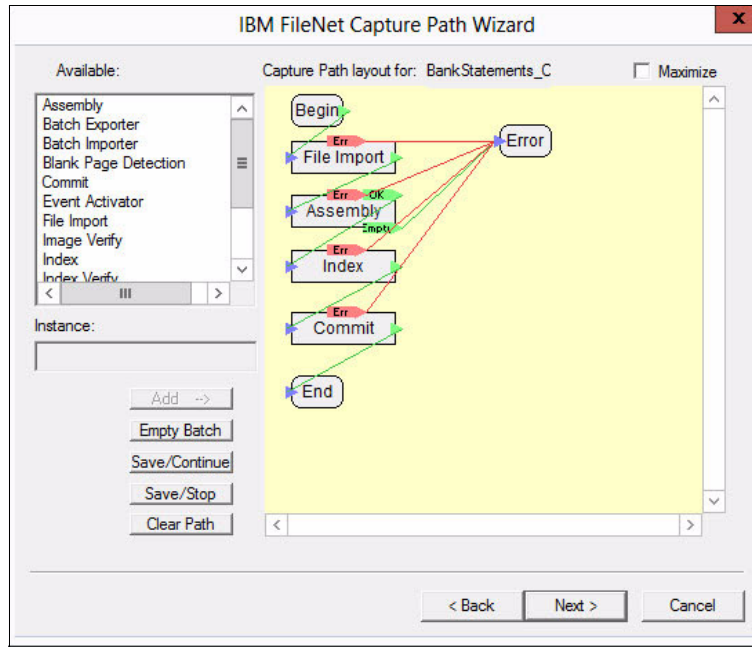A capture path for processing the scenario looks similar to Figure 2.

Figure 2   Capture path

Requirement: One requirement that the bank must achieve is to find and capture form data in varying locations. The capture path in Figure 2 does not contain an Optical Character Recognition (OCR) component. FileNet Capture supports Zonal and Full-text OCR. Neither method enables the bank to meet the scenario requirement: *Automatically find and capture form data in varying locations.*

Characteristics of a capture path

A sampling of capture path characteristics follows:

► Components must be installed on any workstation on which they run.
► Capture paths are stored in a repository.
► Capture paths are created by dragging different components onto a canvas.
► Components in a capture path are selected from a controlled list.
► Batches travel in a linear fashion unless an exception occurs.
► Components can be run in either ad hoc or automated mode.
► Component behavior that is needed beyond what is provided must be written.

Settings collection overview and configuration

A *settings collection* is a collection of properties. Figure 3 through Figure 8 on page 8 shows the properties of each component within the capture path, starting with File Import.

The File Import component (Figure 3 to Figure 5 on page 7) imports files into a batch for processing, satisfying the scenario requirement: *Ability to acquire forms in an electronic format*.
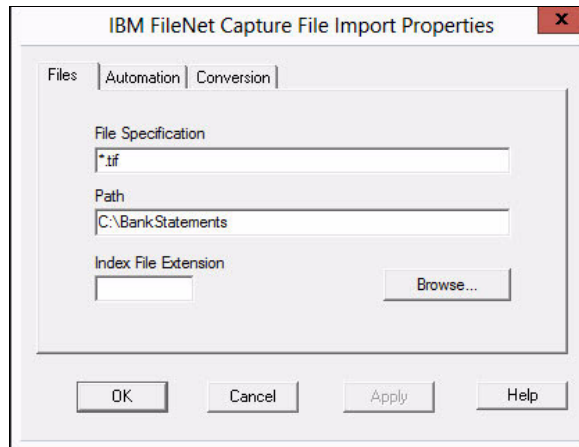

Figure 3   File Import Properties - Files

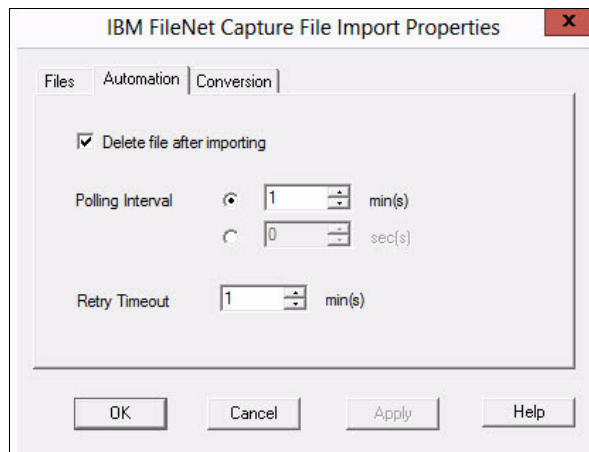Figure 4 shows the File Import Properties - Automation dialog box.

Figure 4   File Import Properties - Automation

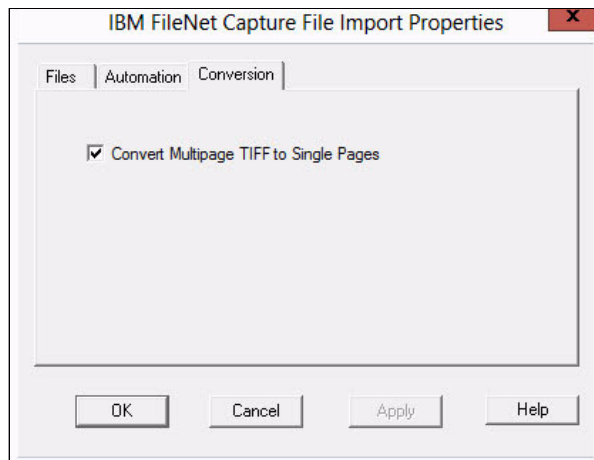Figure 5 shows the File Import Properties for Conversion dialog box.

Figure 5   File Import Properties - Conversion

Assembly, the next capture path component, sorts, orders, and groups individual pages into documents (Figure 6).
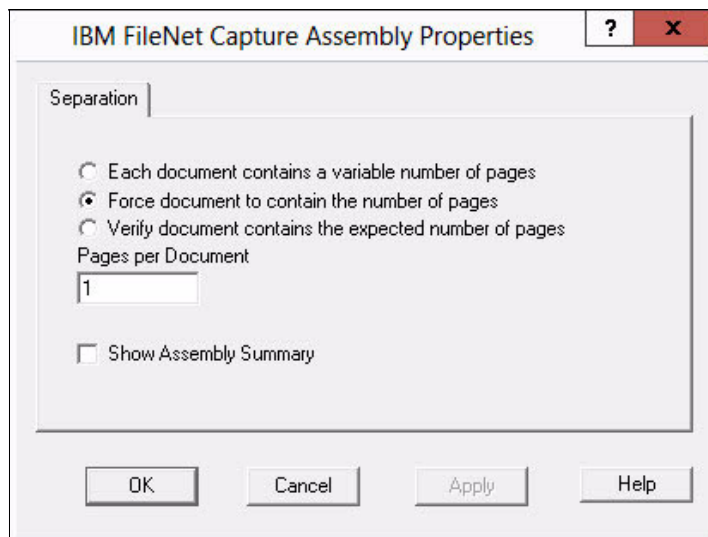

Figure 6   Assembly Properties

Following assembly, the index component is responsible for applying property values to a document for retrieval after it is committed to a repository (Figure 7).
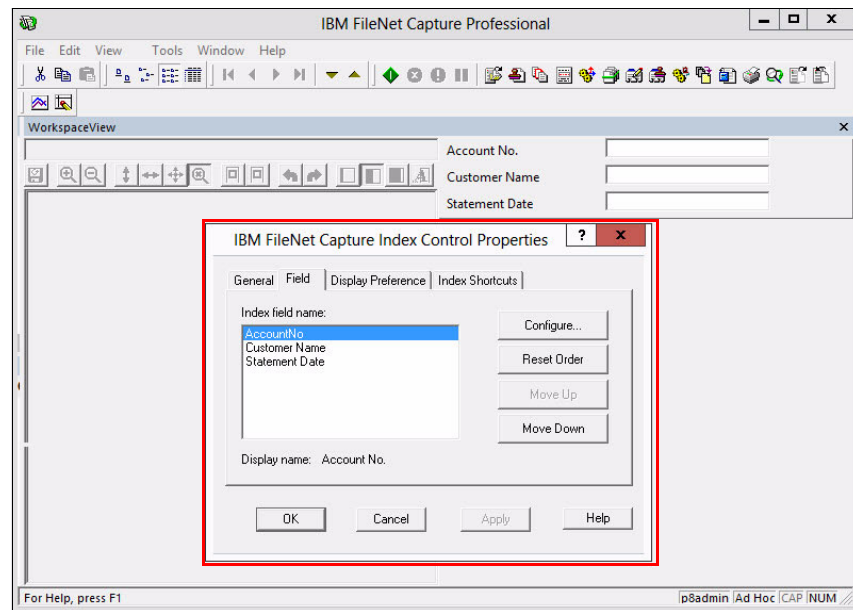

Figure 7   Index Control - Field properties

The final component is Commit. It defines how documents are added to a server or content repository (Figure 8). Commit works with IBM Image Services (IS), IBM Content Services (CS), and IBM Content Engine repositories (FileNet P8). This partially fulfills scenario requirement: *Commit form and form data to any number of content repositories*.

Figure 8   Commit - Content Engine properties

Characteristics of a settings collection

A sampling of settings collection characteristics follows:

► A unique settings collection is created for each document class.

► A single settings collection is required for each variation of settings.

► This information is stored in an IBM Image Services, IBM Content Services, or IBM Content Engine content repository.

► Each setting consists of a fixed list of properties.

Template overview and configuration

To begin processing a batch of work an operator, select a template.

A template is a combination of one or more settings collections and, optionally, a capture path. For automated document capture, they provide a way of specifying the capture path and settings collection an operator uses to process work. For ad hoc document capture, an operator specifies only the settings collection.

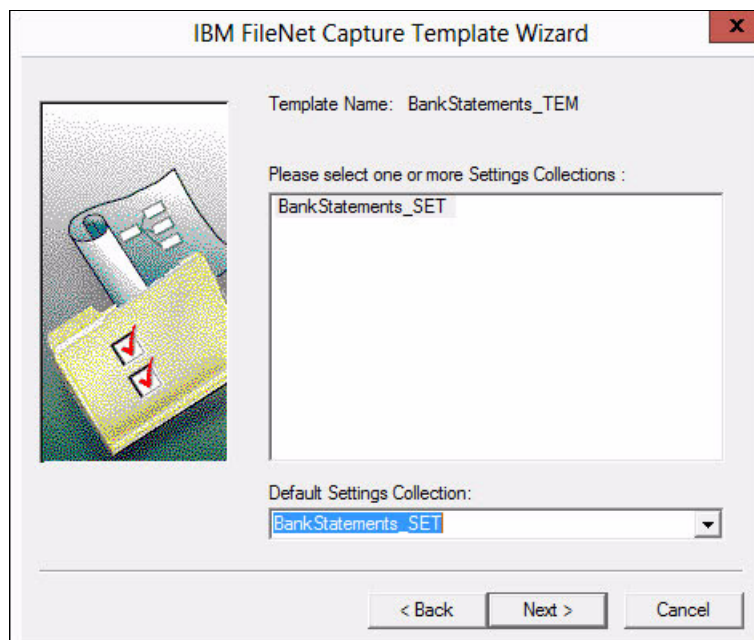Figure 9 shows the FileNet Capture template creation wizard.

Figure 9   Template creation wizard

Characteristics of a template

A sampling of template characteristics follows:

► An operator selects one from a controlled list to start processing a batch.
► You define a relationship between a settings collection and capture path (optional).

Indexing user interface overview

When a batch of work has reached the index component, prior components have populated and validated auto-populated index values.

The index task requires user involvement. An interface opens for addressing and resolving different exception scenarios or allow for manual indexing by an operator. Exceptions might include index validation failure, an index value missing, or visual verification of an index value is required. These conditions require an interface similar to Figure 10.
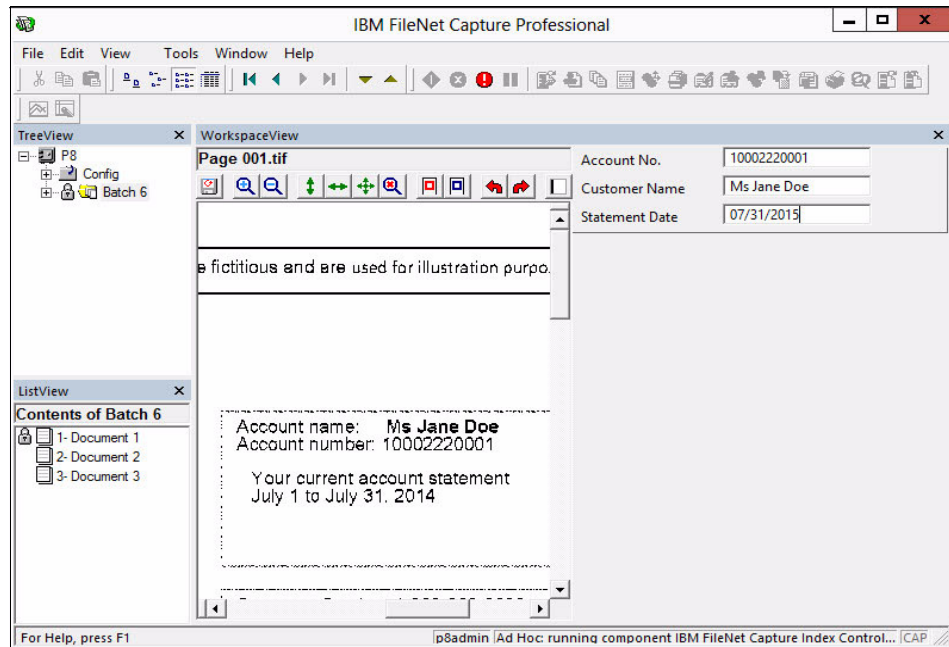

Figure 10   Indexing interface

> **Batch processing:** When processing batches for this scenario, all batches of work stop for an operator to perform manual indexing. This situation occurs because FileNet Capture cannot meet the data capture requirement. The operator must search the page and manually enter the value for each index value.

Characteristics of the indexing interface

A sampling of indexing characteristics follows:

- ▶ Displays indexes that are associated with the selected settings collection.
- ▶ Enforces index validation rules that are defined in the index component properties.
- ▶ Index values can be manually entered or automatically populated.

FileNet Capture processing scenario capability summary

Table 1 lists the scenario requirements and the ability of FileNet Capture to meet them.

Table 1   Scenario processing requirements

Requirement	Yes	No	Partial
Ability to scan paper and index by using a web interface		X	
Ability to scan paper and index by using a desktop interface	X		
Ability to scan paper and index by using multifunctional devices		X	
Ability to acquire emailed forms		X	
Ability to acquire faxed forms	X		
Ability to acquire forms in electronic format	X		
Automatically capture form data that is in varying locations		X	
Validate captured form data			X
Route problematic forms to SMEs for resolution			X
Manage all batches in the capture system throughout their lifecycle			X
Provide access to real-time batch and user processing metrics	X		
Commit form and form data to any number of content repositories			X

IBM Datacap configuration overview

This section outlines how Datacap is configured to meet the scenario requirements. To create a Datacap application, you focus primarily on four areas:

► Task profile
► Workflow
► Processing logic (application)
► Index user interface

Unlike FileNet Capture, when you create a Datacap application, a pre-built template based on preferred practices is applied. With this approach, you start creating your application on a solid foundation.

Task profile overview and configuration

The building blocks of a Datacap application are the creation of task profiles. A task profile contains a collection of rulesets. Each ruleset defines one or more rules that might run against documents, pages, fields, or an entire batch. A rule can contain one or more actions, which perform a specific operation, such as form identification, optical character recognition (OCR), and export.

The first activity that the application performs is to ingest bank statements for processing. Figure 11 shows the ruleset configuration for the ingestion of bank statements. This configuration satisfies scenario requirement: *Ability to acquire forms in an electronic format.*
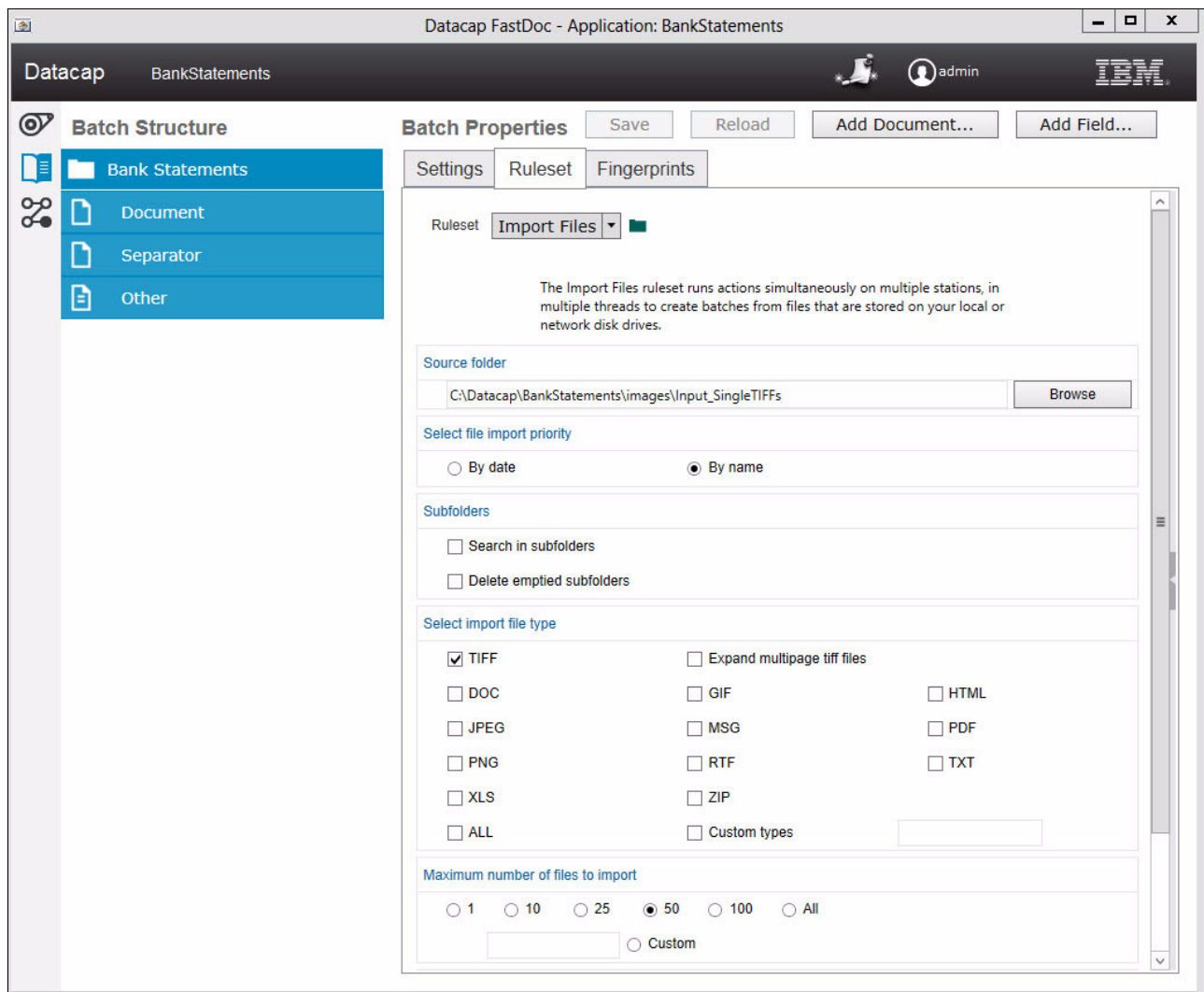

Figure 11   Import Files ruleset

Building rulesets: Datacap provides multiple ways to build rulesets. Figure 11 is the pre-compiled Import Files ruleset. As with all pre-compiled rulesets, many options exist. If a configuration option does not exist, you can add this function by using native actions or write your own option.

When a batch of work is ingested, Page Identification determines the type of page being processed. Multiple automated methods are available, including fingerprinting, structure-based identification, text matching, and manual page identification.

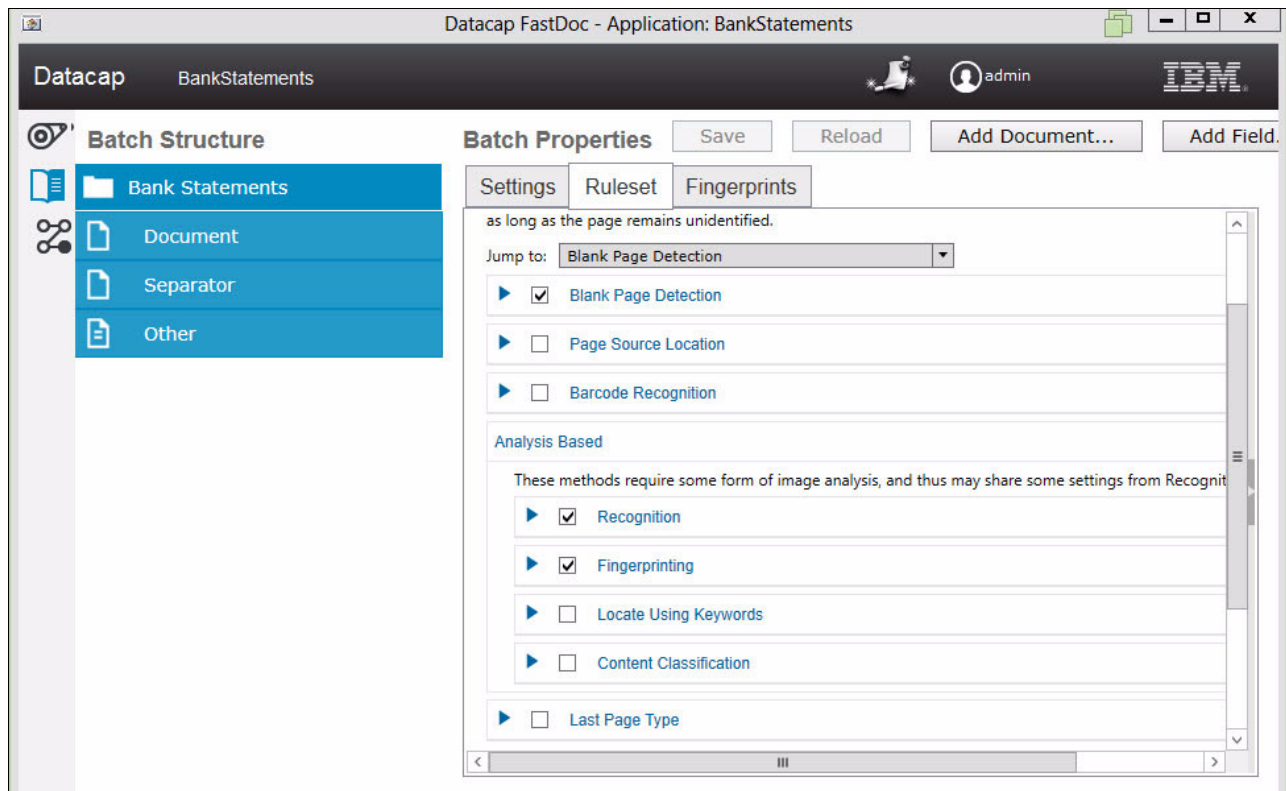Figure 12 show the setting of the image identify ruleset.

Figure 12   Identify Page ruleset - image analysis

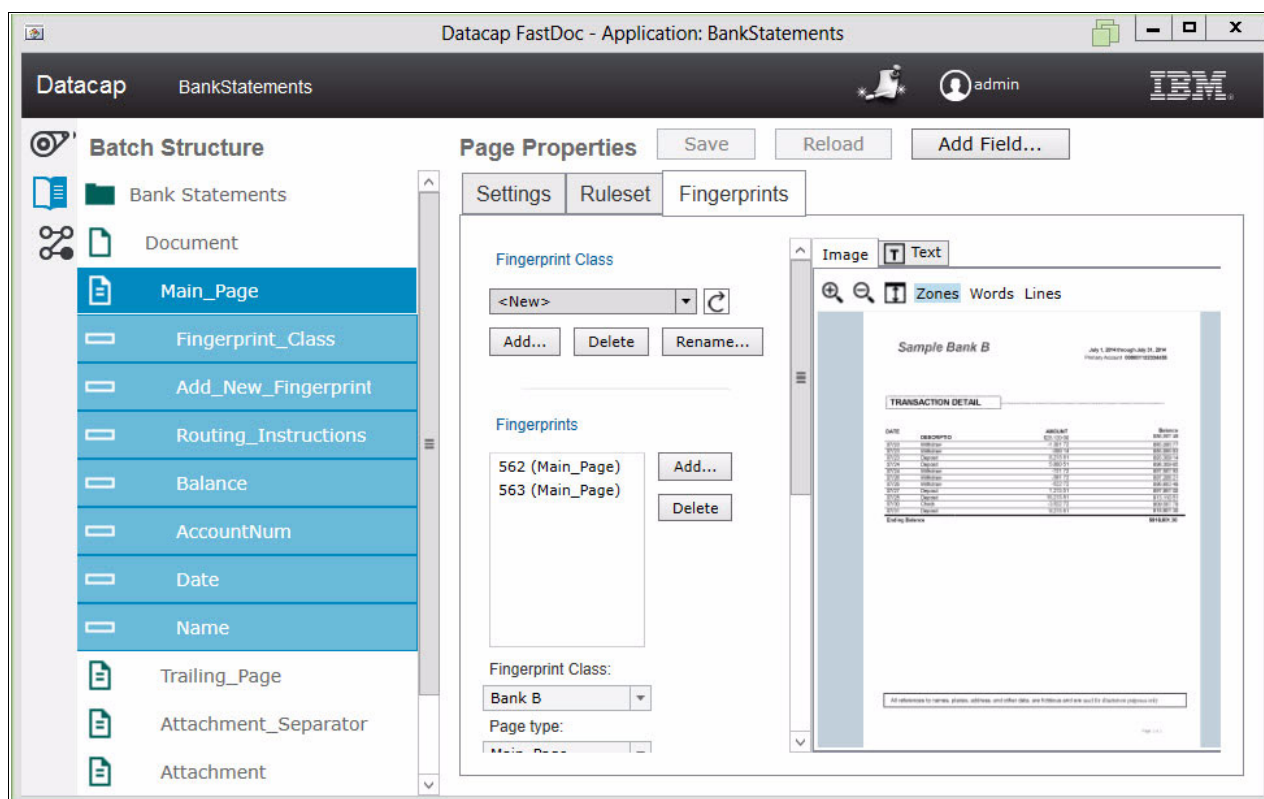Figure 13 shows the Bank Statements page properties ruleset.

Figure 13   Identify Page ruleset - Bank Statements

The Locate ruleset finds data on the page regardless of its location (Figure 14). This approach enables the bank to meet the requirement: *Automatically find and capture form data in varying locations*. Data is acquired by performing recognition on a page. There are several techniques that you can then use to find data, including keyword or pattern matching.
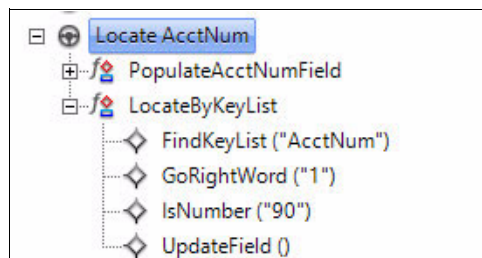

Figure 14   Locate ruleset

The bank wants to validate any data that is extracted from the form. Data validation determines whether captured data conforms to specified business rules, such as length constraints or format expectations, fulfilling the requirement: *Validate captured form data*. Figure 15 shows the Datacap Validate Fields ruleset configuration interface.
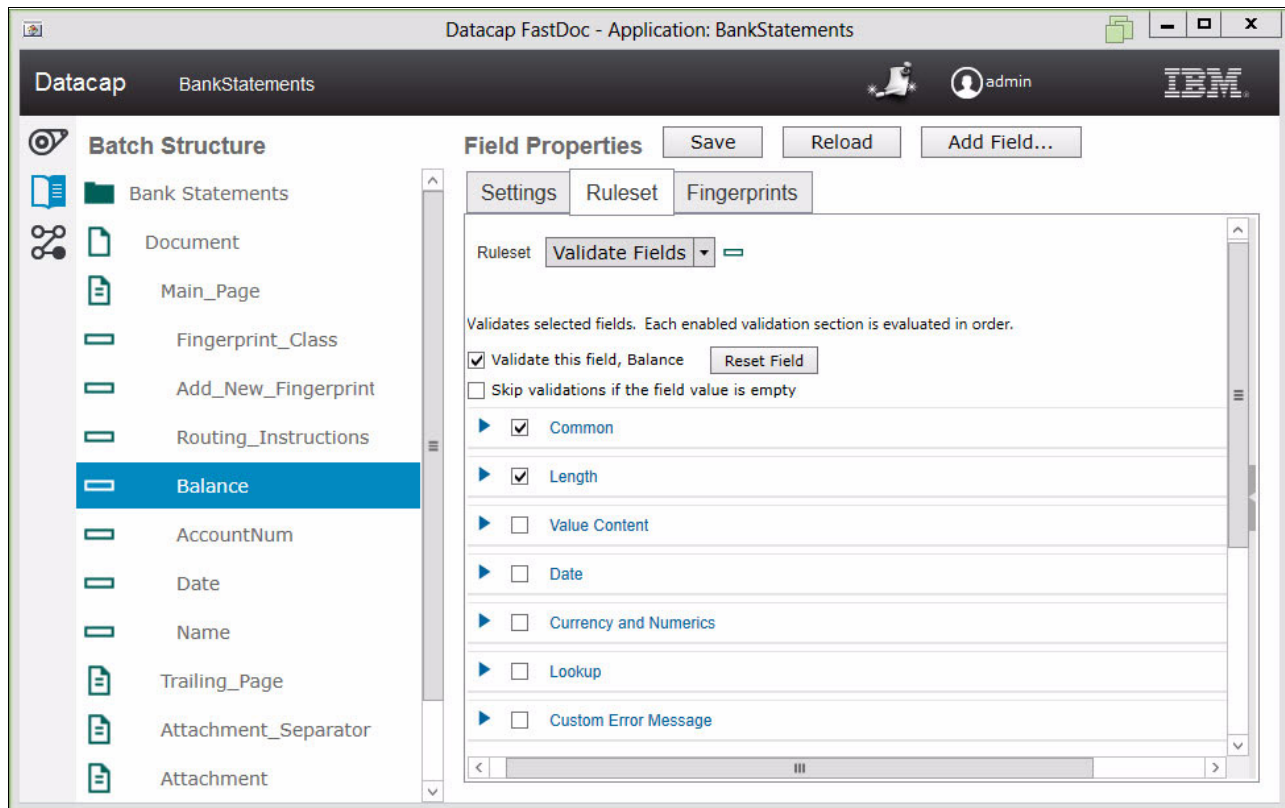

Figure 15   Validate Fields ruleset

To extract data (that fails validation) or to perform manual indexing, an operator uses an index user interface. When IBM Content Navigator is used as the index interface, you can customize its look, feel, and behavior through an intuitive interface (Figure 16). Datacap also provides a desktop index interface.
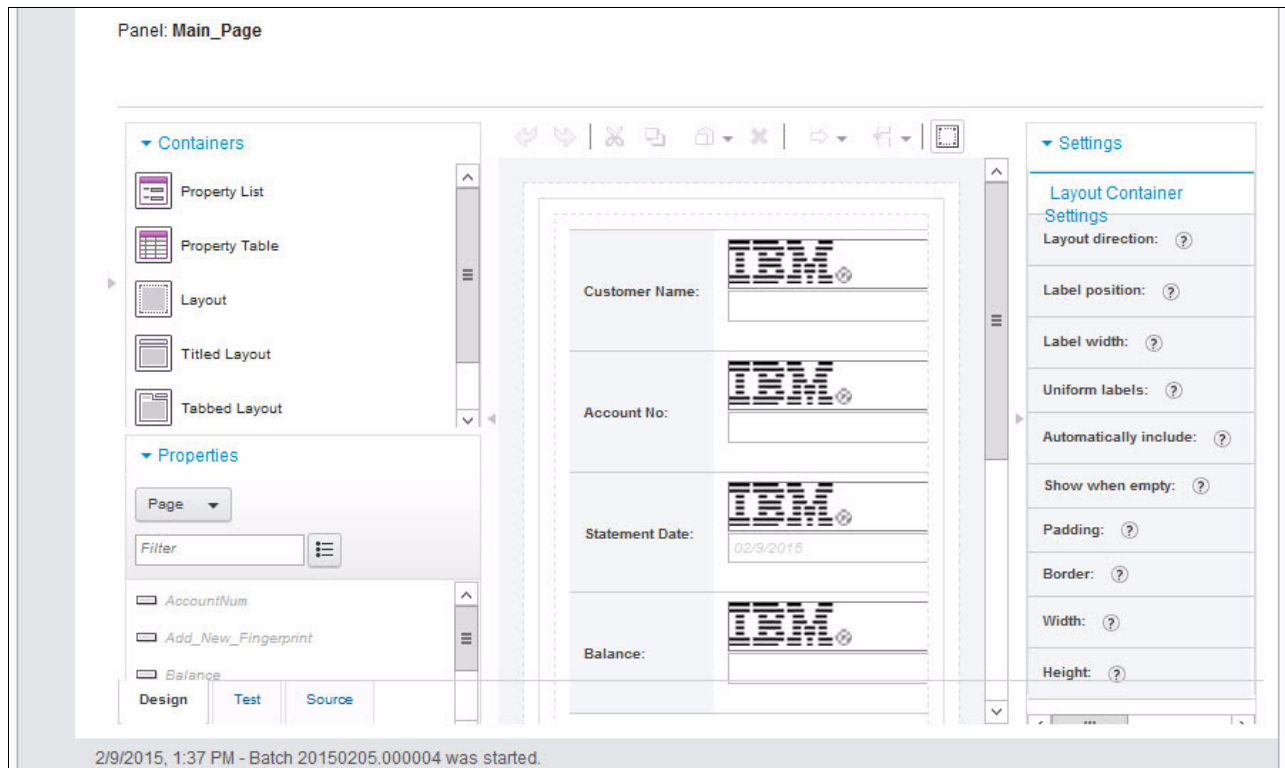

Figure 16   IBM Content Navigator index window creation interface

These interfaces support the following scenario requirements for both scanning and indexing:

► *Ability to scan paper and index by using a web interface*
► *Ability to scan paper and index by using a desktop interface*

The final task is to export the content and documents. Datacap can export data to various receivers, such as a text file, an XML file, a database, and a content management system. The bank can fulfill the requirement *Commit form and form data to any number of content repositories* because of the flexibility of the export options. Figure 17 shows the Datacap export ruleset window.

Figure 17   Export to FileNet Content Manager ruleset
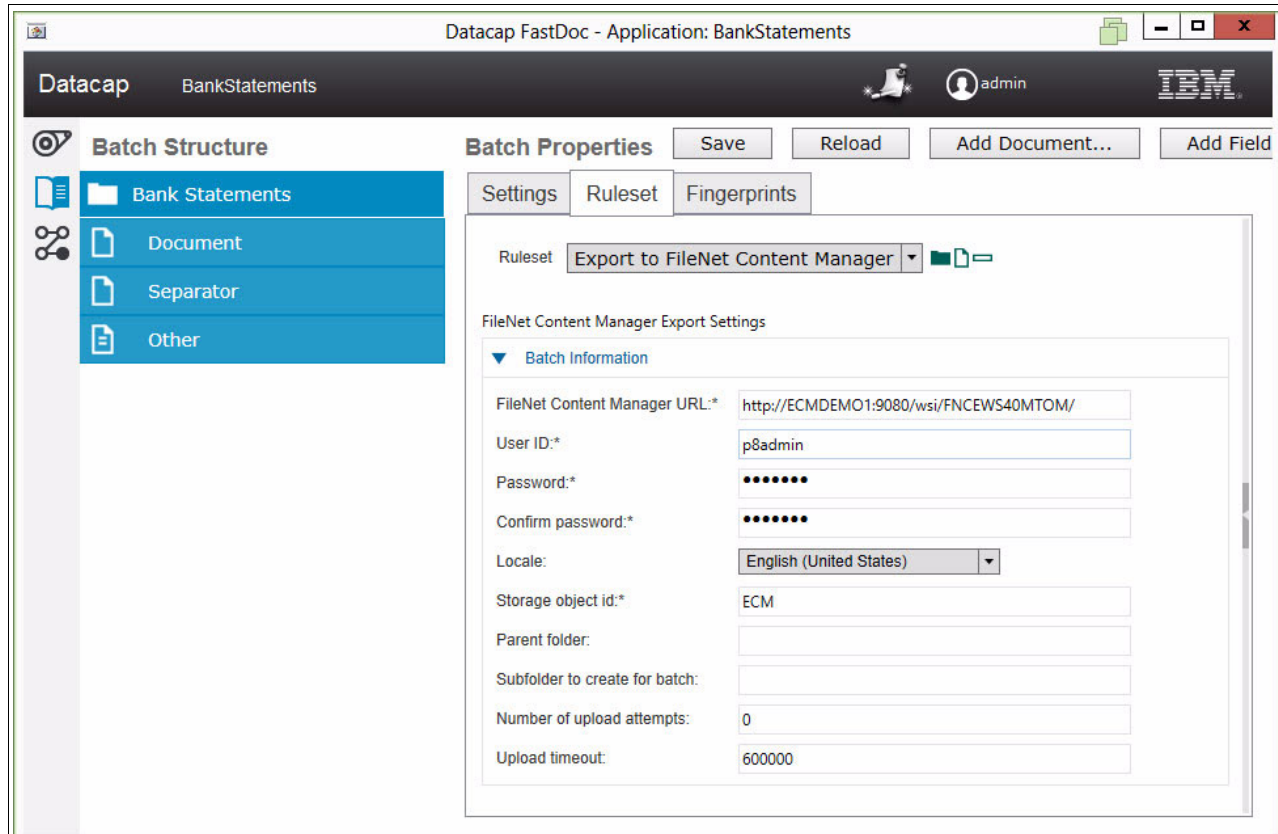
Characteristics of a task profile

A sampling of task profile characteristics follows:

▶ There is no limit to the number of rulesets that can be included in a task profile.
▶ There is no custom coding is required to modify a rule when you use native actions.
▶ It provides the ability to write custom actions that can be reused.
▶ It provides a way to group rulesets.

Workflow overview and configuration

Figure 18 displays a workflow to process bank statements. The workflow is created by dragging rulesets onto task profiles. Through routing logic rules, a batch of work can move to any part of a workflow at any point during the capture process. This ability fulfills the scenario requirement *Route problematic forms to SMEs for resolution*.
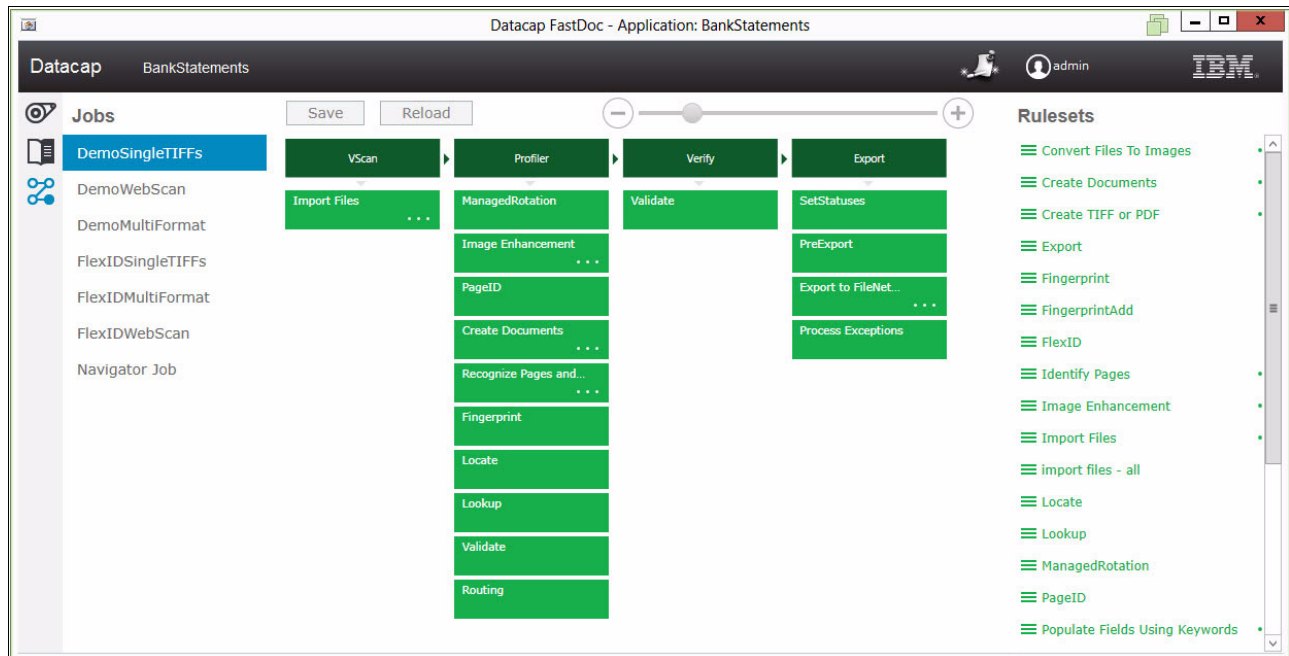

Figure 18   FastDoc workflow creation interface

Characteristics of a workflow

A sampling of workflow characteristics follows:

► You create a workflow definition by dragging rulesets.
► Multiple workflows are automatically created based on preferred practices.
► Batches are processed in a non-linear fashion.
► There is no limit to the number of rules that can be configured for each task.

Application overview and configuration

The Datacap activities that have been described to this point exist in a Datacap application. Applications are designed to ingest, process, and verify the data in your documents. An application is the container that encompasses the logic to perform these types of activities. Additionally, it contains the definitions of the documents it is built to process and how Datacap processes each element within the document.

An application can be created and configured by using multiple interfaces, including both browser- and desktop-based. One desktop interface is FastDoc. With FastDoc, you can build an application and set up a document hierarchy by using an intuitive and straightforward application development interface.

Figure 19 shows the FastDoc interfaces that are used to configure the application to process bank statements for our scenario.
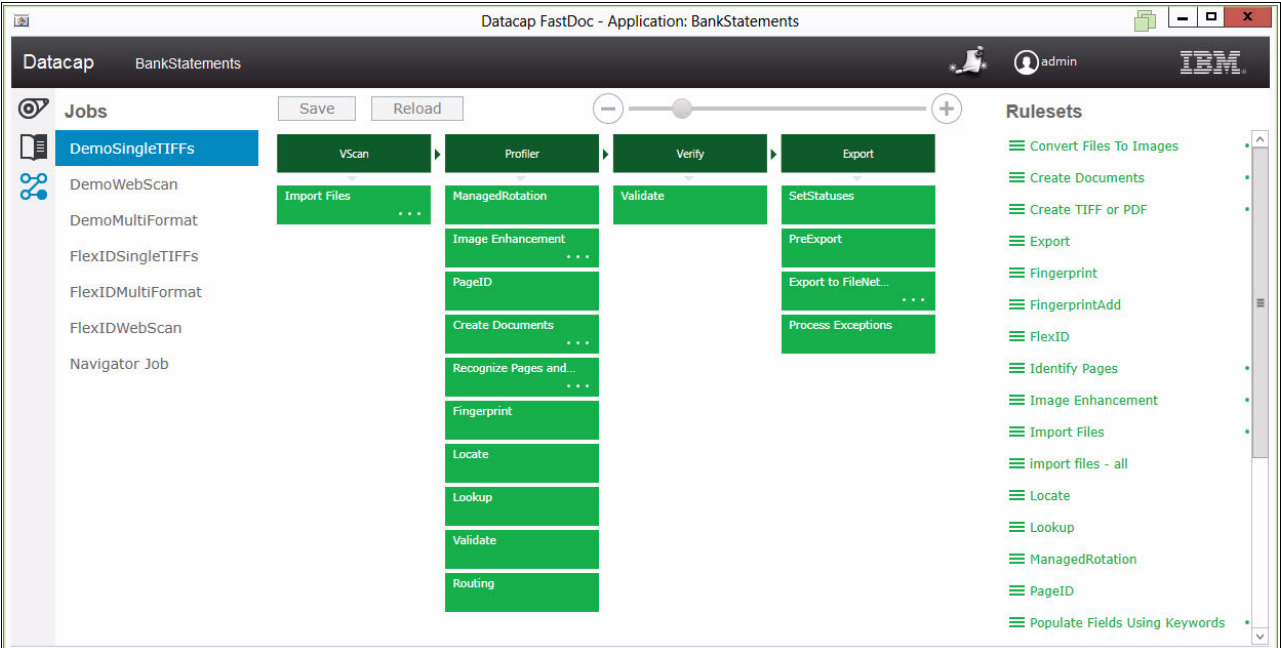

Figure 19   FastDoc Configure Workflow interface

For an overview of a Datacap application workflow, see "Workflow overview and configuration" on page 18.

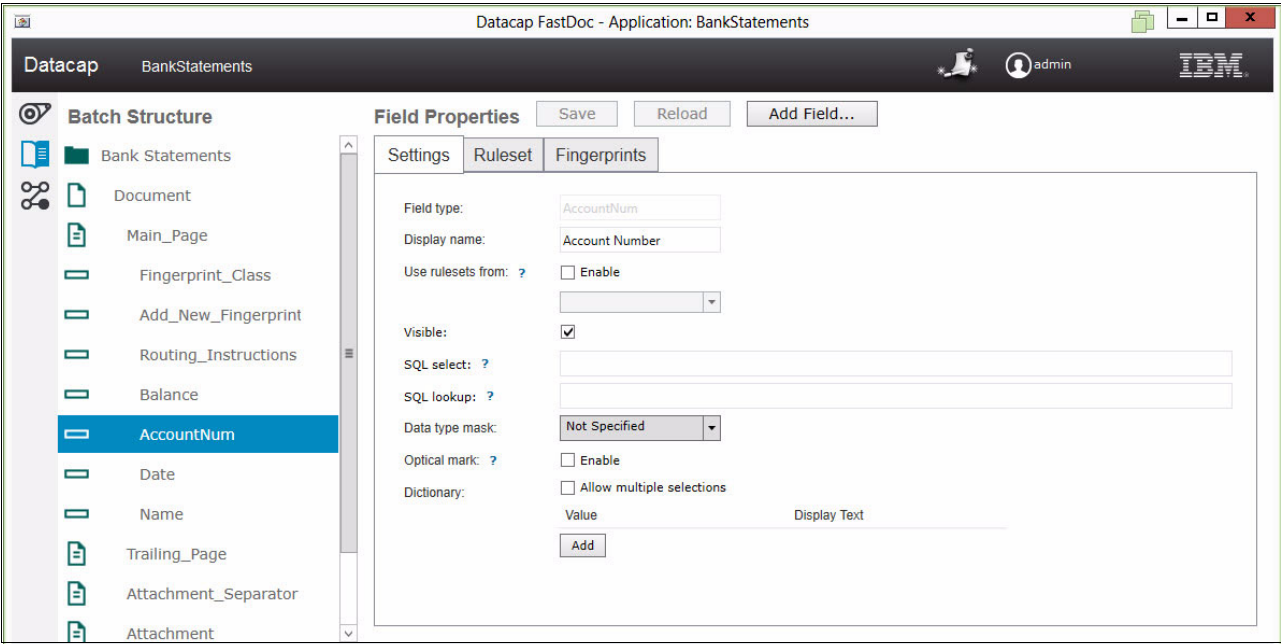Figure 20 shows the FastDoc configuration interface for documents, pages, and fields.

Figure 20   FastDoc configuring documents, pages, and fields interface

Figure 21 shows the FastDoc interface that is used to define the document structure and how each element is processed.
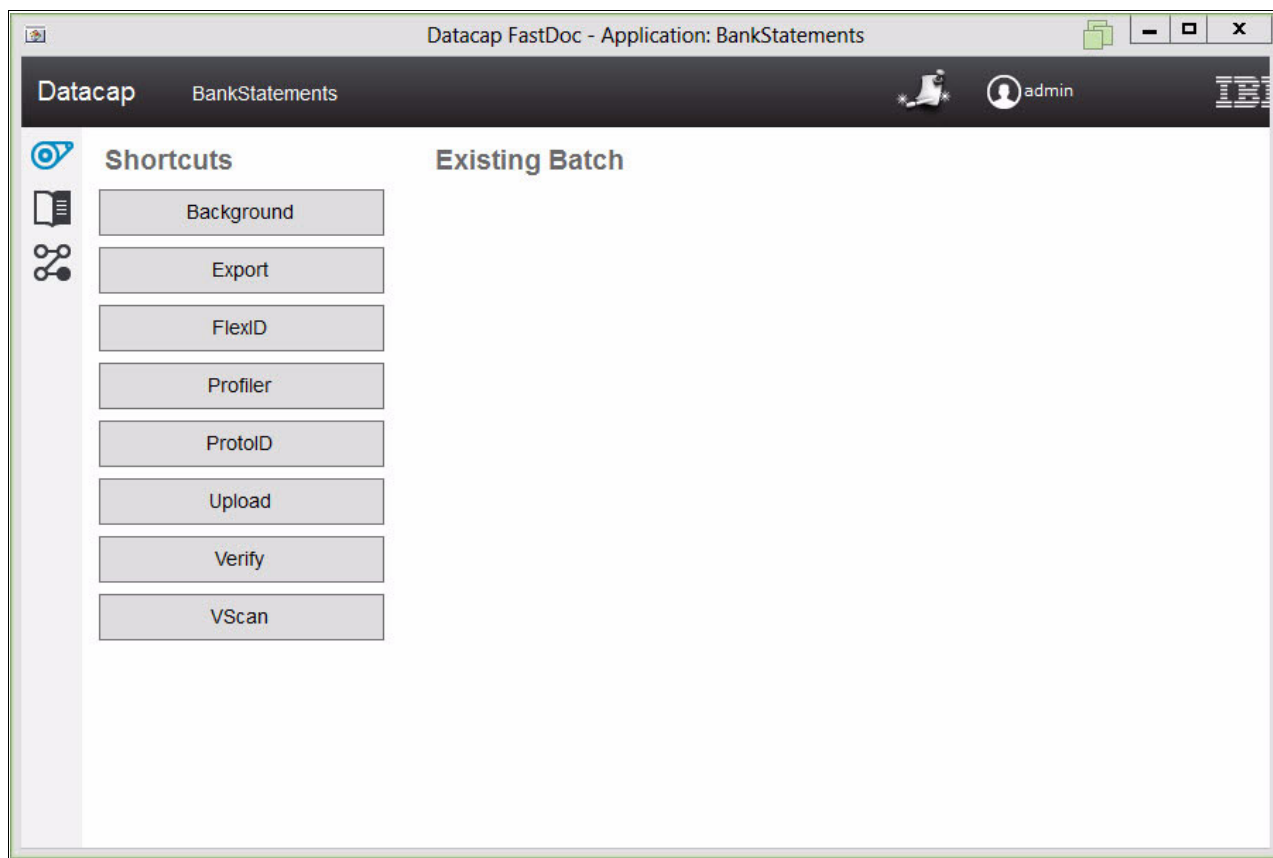

Figure 21   FastDoc Process batches of document interface

With FastDoc, you can process batches for testing while creating an application. A batch can be manually or automatically processed through the capture workflow.

Characteristics of an application

A sampling of application characteristics follows:

► There is a container encompassing the processing logic.
► You can reuse rules across Datacap applications.
► Testing and debugging works within the application configuration environment.
► Defining types of documents the application processes.
► Defining page types that are associated with each document type.
► Defining rules that determine whether the captured data is valid.
► Determining how to route documents that have problems to resolve.
► Deciding how to export or release the data at the end of the capture process.

Index overview and configuration

Indexing is a task requiring user involvement. An interface is displayed for addressing and resolving different exception scenarios or allow for manual indexing by an operator. Datacap offers both desktop and web indexing interfaces to handle these conditions.

The IBM Content Navigator browser provides a drag index creation window interface, as shown in Figure 22.
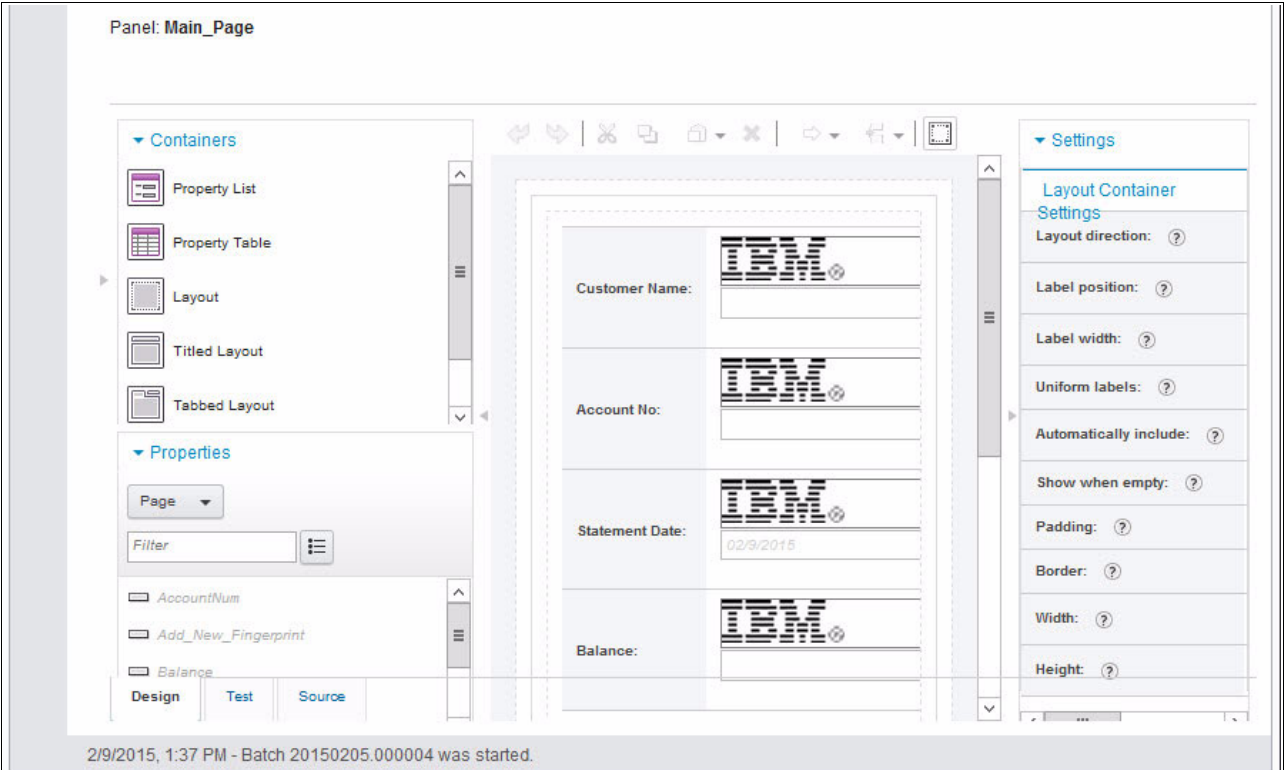

Figure 22   IBM Content Navigator Datacap index creation window

Additionally, the Datacap Desktop is a desktop application that can be used to perform indexing. Figure 23 shows a Datacap Desktop indexing interface.
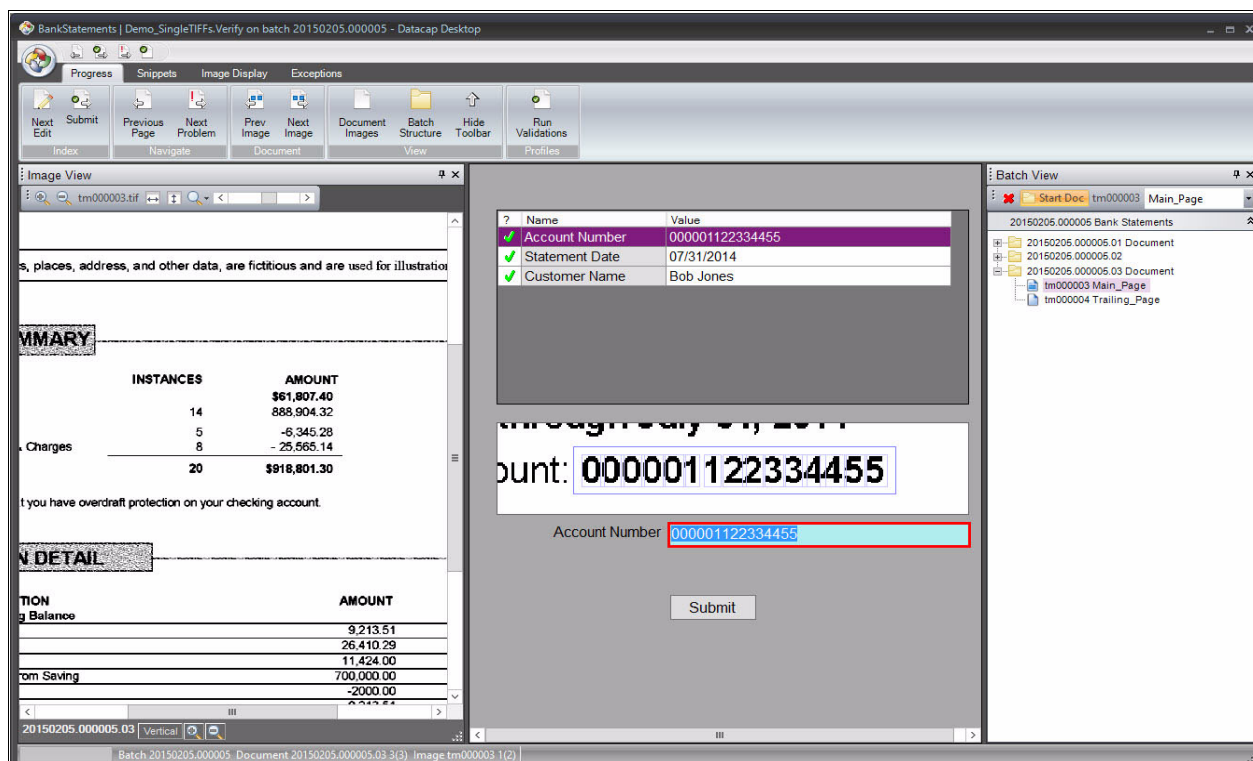
Figure 23   Datacap Desktop indexing interface

Characteristics of an index interface

A sampling of indexing characteristics follows:

► Real-time data validation
► Real-time database lookup abilities
► Multiple pre-created index interfaces
► Browser-based index creation window
► Support for manual indexing

Datacap processing scenario capability summary

Table 2 lists the scenario requirements and the ability of Datacap to meet them.

Table 2   Scenario requirement and Datacap capabilities summary

Requirement	Yes	No	Partial
Ability to scan paper and index by using a web interface	X		
Ability to scan paper and index by using a desktop interface	X		
Ability to scan paper and index by using multifunctional devices	X		
Ability to acquire emailed forms	X		
Ability to acquire faxed forms	X		
Ability to acquire forms in electronic format	X		

Automatically capture form data that is in varying locations	X		
Validate captured form data	X		
Route problematic forms to SMEs for resolution	X		
Manage all batches in the capture system throughout their lifecycle	X		
Provide access to real-time batch and user processing metrics	X		
Commit form and form data to any number of content repositories	X		

Characteristics of an advanced capture solution

As technology advances and methods to receive content grow, so do the requirements for businesses to adapt to these changes. To accommodate these changes, businesses must have an advanced capture solution. This section outlines the high-level concepts that an advanced capture system offers.

An advanced capture solution provides you with the tools and flexibility to ensure that the following goals are met:

► Monitor and ingest from multiple content sources

► Provides multiple advanced techniques to identify forms

► Automatically detects and resolves common processing issues

► Intelligently locates content regardless of its location within a form

► Validate manually and automatically captured values against external systems

► Provides multiple, flexible scan and indexing interfaces

► Exports captured content to both IBM and non IBM systems

► Provides immediate access to capture processing metrics

Capture process note: At any point in the capture process, throughput and efficiency decrease when a person becomes involved. Therefore, the only time a person should become involved is when there is no other option.

Multi-channel input

Before any capture process begins, content must be ingested into the system. As technology advances, so do the various content input channels. Today, a capture solution must integrate with many different channels to ensure processing efficiency and to meet customer requirements. This can be both standard and custom content sources:

► Email servers
► Multifunction devices (MFDs)
► File shares
► Content repositories

Form identification

After content is ingested, regardless of the source, it is identified. An advanced capture solution must use any number of techniques to perform this critical step:

► Detection of keywords or phrases
► Identification of logos
► Identification based on the location of a page within a batch
► Amount of content on a page
► Analyze page content

Data extraction

When a page is identified, a system must be able to read the content from the page. The most basic form of extraction, which requires content to always be in the same location, is often referred to as *zonal OCR*. In this case, a zone is drawn around the value on a form during the setup and configuration stage. Although this works in some cases, more forms have content in different locations. A capture system's inability to find content, regardless of its location, results in all batches having to be manually indexed. To eliminate this problem, advanced extraction techniques are critical. These techniques include the following functions:

► Find data by using advanced search techniques, such as regular expressions and keywords
► Detect anchor words and capture surrounding words (for example, above and below)
► Detect and read any number of line items

Data validation

Data validation is a crucial step in ensuring that any automatically extracted or manually keyed values are valid. Capturing and populating invalid data in downstream systems can result in bad decisions being made or misfiled content. To ensure that action does not occur, an advanced capture system must provide the ability to create robust and flexible validation rules. Key concepts surrounding data validation must include the following items:

► The ability to use an external system during the validation process
► The ability to create flexible and reusable validation rules

User interfacing

The quality of form identification, data extraction, and data validation is highly dependent on the quality of the image being processed. When the capture system cannot o identify, extract, or validate parts of the document, a user interface is necessary to present cleanly and efficiently the exception to a user for resolution. A user interface must adopt to accommodate the following characteristics:

► Allow arrangement of the index fields in a logical manner
► Provide real-time validation feedback
► Extract data from certain forms without using a keyboard

Content export

Key to a capture process is flexibility in methods for placing both extracted content and documents in one or many locations. Extracted content and documents should be able to be treated as a unit or individually. An example is to commit content into a database and the document into a content repository. Additionally, the ability to commit captured content to multiple locations is key when operating within an environment of different systems.

Real-time reporting

For many capture processes, it is critical that they complete in an expedited period. Often, the content being captured is what is used to both make business decisions and drive operations. Many processes are also tied to a revenue stream. The ability to determine the state of a document, audit who processed it, and gain insight into both efficient and inefficient processes ensure that service level agreements (SLAs) are met.

In addition to having access to this content, the manner and speed in which you can access it is important. The most flexible and accessible interface is through a browser. A browser interface ensures access from a smartphone, notebook, desktop, and other mobile devices. Rapid and flexible access to capture processing metrics is essential.

Summary

Capture technologies continue to change, which forces businesses to be flexible. The processing scenario overview in this paper shows that both FileNet Capture and Datacap can process the sample bank statements. This paper also shows that Datacap offers a more adaptable, automated, and efficient solution. This situation can be attributed to the fact that Datacap is an advanced capture solution.

Table 3 shows a side-by-side comparison matrix of the abilities of Datacap and FileNet Capture to meet scenario requirements, and their advanced capture processing abilities.

Table 3   Comparison matrix

Requirement	Datacap	FileNet Capture
Ability to scan paper and by using a web interface	Yes	No
Ability to scan paper and by using a desktop interface	Yes	Yes
Ability to scan paper and by using multifunctional devices	Yes	No
Ability to acquire emailed forms	Yes	No
Ability to acquire faxed forms	Yes	Yes
Ability to acquire forms in electronic format	Yes	Yes
Automatically find and capture form data in varying locations	Yes	No
Validate captured form data	Yes	Partial
Route problematic forms to SMEs for resolution	Yes	Partial
Manage all batches in the capture system throughout their lifecycle	Yes	Partial

Provide access to real-time batch and user processing metrics	Yes	Yes
Ability to monitor and ingest from multiple content sources	Yes	Partial
Provide multiple advanced techniques to identify forms	Yes	No
Validate manually and automatically captured values against external systems	Yes	No
Export capture content to both IBM and non IBM systems	Yes	Partial

This paper reviewed the multiple objectives that are centered around the IBM FileNet Capture and IBM Datacap products and reviewed the capabilities of an advanced capture solution. The paper has hopefully provided insight into both of these products and their capabilities.

Authors

This paper was produced by a specialist working at the International Technical Support Organization, San Jose Center.

Kevin Bowe is an IBM Client Technical Specialist in the Enterprise Content Management space for over 20 years. His experience includes business process reengineering, system architecture, project management, release and regression testing, development, proposal creation and response, presentations, demonstrations, and hands-on experience with advanced capture and capture products. Most of Kevin's Enterprise Content Management experience is derived from the IBM FileNet product suite.

Thanks to the following people for their contributions to this project:

Whei-Jen Chen and LindaMay Patterson
International Technical Support Organization, San Jose Center

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience by using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Stay connected to IBM Redbooks

- ► Find us on Facebook:

  http://www.facebook.com/IBMRedbooks

- ► Follow us on Twitter:

  http://twitter.com/ibmredbooks

- ► Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806

- ► Explore new IBM Redbooks® publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm

- ► Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

This document REDP-5236-00 was created or updated on July 6, 2015.

Send us your comments in one of the following ways:
- ► Use the online **Contact us** review Redbooks form found at:
  ibm.com/redbooks
- ► Send your comments in an email to:
  redbooks@us.ibm.com
- ► Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400 U.S.A.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

Redbooks (logo) ®	IBM®	Redpaper™
FileNet®	Redbooks®	

The following terms are trademarks of other companies:

Other company, product, or service names may be trademarks or service marks of others.

Get connected

ibm.com/redbooks