EMC® Input*Accel*®

Version 6.0

System Overview: The Basics of InputAccel

P/N 300-007-692

EMC Corporation
Corporate Headquarters:
Hopkinton, MA 01748-9103
1-508-435-1000
www.EMC.com

Document number: systemoverview — 6.0 (VC 10–03–2008) enu

Table of Contents

Chapter 1

Overview

InputAccel is a web-based, global input management system that converts business critical information from paper or electronic sources into digital content and delivers it to back-end systems. Immediate access to indexed information promotes expedient business decisions and efficient transactions. By working in a global environment, InputAccel reduces transmission costs and takes advantage of time zone differences and low network traffic.

InputAccel is a scalable solution that uses multiple servers to manage all available resources and processes at the batch, document, or page level. When one server shuts down, the other servers continue batch processing. Since pages can proceed to the next task without waiting for the entire batch to finish processing, InputAccel decreases processing time and increases throughput to back-end systems. By reducing manual data entry, InputAccel minimizes processing errors, improves data accuracy, and boosts productivity. InputAccel:

- Reduces operating costs caused by factors such as document preparation and data entry.
- Reduces paper costs caused by mishandled physical documents.
- Improves information quality for critical business processes.
- Accelerates business processes by providing immediate access to all information and supporting documentation.
- Enforces stronger compliance control by storing electronic documents.

Topics in this section include:

Understanding information capture

InputAccel converts information captured from printed documents, faxes, and email messages into digitized data and stores the data and images into back-end systems for fast and efficient data retrieval. An InputAccel process:

- **Captures information**
  - Captures paper, faxes, film, images, or imported electronic documents (structured and unstructured) through fax, scanner, network drives, and remote sites.
  - Improves image quality, cleans up images to improve image clarity and readability, and speeds up processing-without manual intervention.
  - Enhances images to improve recognition results and organizes multi-page documents into document sets.
- **Extracts information**
  - Performs optical or intelligent character recognition (OCR) to extract machine and handprint text using zonal OCR for structured documents and full-text OCR for unstructured documents.
  - Reads bar codes to extract alphanumeric data.
  - Enables key from image data processing.
  - Populates data from an external source using scripting events.
- **Validates information**
  - Maintains data integrity using restriction masks, regular expressions, and numeric only field properties.
  - Validates data against an external database or custom business rules using scripting events.
- **Delivers information**
  - Exports both images and index data to leading content management systems, ERP, BPM, databases, and other systems.
  - Supports conversion to PDF, full-text OCR, and PDF compression.

Understanding client/server technology

InputAccel runs in a client/server network environment. An InputAccel Server is an open integration platform that manages and controls the document capture process by routing document pages and processing instructions to the appropriate client modules.

Multiple InputAccel Servers can be licensed and configured into a ScaleServer group, which acts as a single information capture system. ScaleServer-compatible modules can connect to multiple InputAccel Servers in a ScaleServer group and receive tasks from all of them. Modules that are not ScaleServer-compatible can connect to one InputAccel Server at a time within a ScaleServer group.

In the Process Developer module, you program an Integrated Process Project file (IPP) in VBA that instructs the InputAccel Server which modules to use, the order in which to use them, how each module needs to process pages, and what to do with the data generated by each module. Compiled Integrated Process Project files have an IAP extension. Each batch file contains an embedded IAP file that enables the InputAccel Server to perform workload balancing, keep track of where each task belongs in its respective batch, and save each image and text file generated during each step of processing. Because each task is self-contained, modules can process tasks from any batch in any order. The interaction between InputAccel Servers and modules enables asynchronous task processing. The modules can process tasks from any batch without regard to which pages go with which document capture job. Independent task processing increases throughput and decreases processing time.

InputAccel client modules are software modules that perform specific information capture tasks such as scanning images, enhancing images, performing OCR, indexing data, or exporting images and data. InputAccel supports third-party InputAccel certified modules. All modules running on client workstations use TCP/IP, an industry standard protocol, to connect to the InputAccel Server.

Over the Internet, InputAccel uses Web Services components to participate in the Service Oriented Architecture (SOA) environment, which enables businesses to exchange data without regard to their location, operating system, or platform. The Web Services Input and Web Services Output modules communicate over the Internet using Service Oriented Access Protocol (SOAP), which is a protocol for exchanging Extensible Markup Language (XML)-based messages over HTTP/HTTPS.

Using ScaleServer groups

InputAccel ScaleServer technology provides InputAccel systems with many benefits including increased availability, higher productivity, improved workload balancing, and centralized control. In a ScaleServer group, up to eight InputAccel Servers work together as a single information capture system, distributing the processing workload. Each server in the ScaleServer group manages its own work, and each client workstation requests work from all available servers. When a module finishes processing, it sends the batch back to the InputAccel Server where it originated. The multiple servers appear as a single server to the module. Servers share connection information so a module consumes just one connection license.

In a single-server platform, an InputAccel Server must remain constantly available to fulfill storage and processing requirements. When an InputAccel Server fails due to hardware or software issues, all processing stops. In a ScaleServer environment, the loss of a server does not impact processing.

The number of pages that can be processed in a specified time period is controlled by an InputAccel license. Typically, a module that creates batches such as ScanPlus or Email Import requests additional pages from an InputAccel Server. To increase productivity and throughput, InputAccel 6.0 allows individual servers to share pages with other servers in a ScaleServer environment instead of becoming unavailable when their page count allotment runs out. Now, when a specific InputAccel Server needs additional pages, it requests them from another server in the ScaleServer group. The server keeps asking until the request is satisfied. If the request cannot be satisfied, then a message displays a warning regarding low page counts. The InputAccel Servers perform page sharing without impacting the client module.

InputAccel supports side-by-side installation of the same version of InputAccel Servers by specifying a unique name for each InputAccel Server.

Information on licensing, configuring, setting up, and using ScaleServer technology can be found in the Using Administration Console section in the Administration Guide.

Chapter 2

A quick look at InputAccel modules

InputAccel 6.0 includes new and updated modules that offer the latest technologies and enhance the capabilities of the existing modules. Updated modules have been rewritten with new features and an enhanced user interface. Refer to the Installation Guide for a list of supported modules and upgrade information. The new and updated modules are:

- Administration Console: New module that replaces the Administrator and Supervisor modules.
- Script Engine: New module.
- Documentum Advanced Export: Updated module that replaces Documentum Server Export.
- Email Import: Formerly an EMC Consulting module, that is now a standard installed and supported module.
- Image Divider: Formerly an EMC Consulting module, that is now a standard installed and supported module.
- IndexPlus: New module that replaces the Index module.
- Multi-Directory Watch: Formerly an EMC Consulting module, that is now a standard installed and supported module that replaces the Watch module.
- NuanceOCR: Updated module that replaces theScanSoftOCR module.
- PrimeOCR Plus: Updated module that replaces the PrimeOCR module
- RescanPlus: New module that replaces the Rescan module.
- ScanPlus: New module that replaces the Scan module.
- Web Services Input: New module.
- Web Services Output: New module.

After installing InputAccel, you can access the documentation for each module from **Start > Programs > InputAccel > Documentation > InputAccel Documentation**.

Topics in this section include:

Administration and Development

Use the InputAccel administration and development tools to configure and monitor an InputAccel system, create and install processes, and customize the behavior of modules at run-time.

- Administration Console: Web-based module that allows administrators to monitor, configure, and control an InputAccel system over the Internet. Using Administration Console, an administrator:
  - Configures a ScaleServer group.
  - Performs all InputAccel Server administrative tasks.
  - Controls all process and batch administration, including monitoring batch traffic and finding batches.
  - Performs administrative tasks related to InputAccel modules.
  - Installs and maintains InputAccel licenses.
  - Interacts with the InputAccel Database.
  - Configures the logging subsystem to capture informative, real-time data.
  - Configures and generates informative, customizable reports.
- ECM Web Services Importer Configuration: Configures and manages client steps of the web service. When properly configured, a client step can process requests from software outside an InputAccel document capture system and route those requests into InputAccel as a new batch.
- Process Developer: Provides an integrated development environment using Microsoft Visual Basic for Applications (VBA) for creating, installing, testing, and compiling Integrated Process Project (IPP) files. Although similar to Microsoft Visual Basic, VBA cannot be used to create stand-alone applications. Process Developer resembles the VBA programming environment, but has been modified to work specifically with the InputAccel Server. InputAccel processes (IPPs) are generated from Visual Basic scripts.
- Script Engine: Contains a single interface to write custom code to perform database searches, save data to a repository, manipulate nodes in a tree, and perform other custom functions. Use the Script Engine module to run a script between other steps in a process. When run as a client module, add it to a process (IPP) and then run it in setup mode to configure its script events.

Batch Processing

The majority of InputAccel components are used to directly run batches, enabling you to import, process, and export images and other data.

- Scanning and importing, page 10
- Image processing and annotation, page 11
- Optical character recognition, page 12
- Indexing, page 12
- Export, page 13

Scanning and importing

The following modules import data into the system from physical and electronic sources. Most processes start with one of these modules:

- ECM Web Service Importer: Configures a third-party Enterprise Content Management (ECM) system to route documents into InputAccel for processing and then export them back into the ECM system.
- Email Import: Imports email from a configured mailbox into an InputAccel process by creating a new batch. During processing, the module imports the email body as text into the new batch.
- Multi-Directory Watch: Monitors multiple directories for new image files. When new images are detected, the module creates a new batch file (IAB) based on the IPP defined for the directory where the new images were found.
- RescanPlus: Web-deployable client module that rescans pages that were rejected for quality issues at a page level. Only pages that need rescanning are reprocessed, not the entire batch. Rescanned pages are positioned in their original place in the batch. ScanPlus cannot be used because ScanPlus creates tasks, rather than receives tasks.
- ScanPlus: Web-deployable client module that includes support for over 300 scanners and leading scanner controller boards. In setup mode, you define batch hierarchical structures based on detected events, scanner settings, and scanner operations. Creates batches by importing, opening, or scanning pages.
- Web Services Input: Import module that enables InputAccel to receive and process documents over the Internet. It consists of three components:
  - Web Services Input: Receives data for processing, which includes documents or the URLs of documents, over the Internet and creates batches. Any document type that is compatible with InputAccel can be processed. When the module receives a document, it sends a stage file containing the document to the InputAccel Server. During setup, parameters are mapped to IA Values, which are also passed to the InputAccel Server.
  - Web Services Hosting: Acts as a Web Services server and receives requests for a specific Web Services Input module. In addition, this module extracts input parameters in XML format for subsequent processing by the Web Services Input module, transmits web request parameters to the Web Services Coordinator component, and waits for output parameters to send a response to the caller.
  - Web Services Coordinator: Manages the Web Services Hosting component, maintains the request and response data in the InputAccel SQL database, and provides routing between Web Services Hosting and the Web Services Input module. In addition, Web Services Coordinator maintains task queues and synchronizes their processing.

Refer to the Installation Guide for more information on deploying the ScanPlus and RescanPlus modules using ClickOnce technology.

Image processing and annotation

The following modules enhance, manipulate, and add annotation data to images:

- Auto Annotate: Applies annotations to images and creates new images containing the annotations. Use Auto Annotate to highlight or hide information, draw lines and shapes, stamp each page with a bitmap image, and add text to each page.

- Image Divider: Acquires, identifies, and processes multi-page image files. When Image Divider identifies an incoming file as a multi-page image file, it splits the file into single-page files while preserving the attributes of the original image file. Separating the image splitting functions from other processing tasks is useful within information capture systems that import multi-page files. By isolating the image splitting functionality, the process can be run unattended, freeing the import module of this responsibility.
- Image Enhancement: Cleans up scanned images prior to OCR operations and reduces image file size. During setup, a filter list can be created and then applied to a batch of images in production. Features include automatic deskewing, cropping, registration, filtering, and character repair. The filters increase recognition results by improving image quality. This process can be performed with or without operator input.
- Image Quality Assurance: Performs quality assurance on images. Quality assurance enables operators to correct the orientation of images and to assign an image reject reason such as skewing, too light/too dark, bent corners, or noise. Based on instructions in the IPP, the rejected images route to other modules for further processing. This module also can be used to manually index images using one or more key entry zones.
- Page Registration: Aligns an image with the master image using text and graphic registration zones. When using OCR on structured documents, it is important that all of the scanned images line up with a clean sample form, so data locations are precisely aligned from form to form. This module is designed to process all of the pages in a batch so that all scanned images conform and line up with a clean sample image. Enables users to perform OCR on static form fields with confidence that each zone is associated with the correct area of an image.

Optical character recognition

The following modules perform optical character recognition:

- NuanceOCR: Performs optical character recognition of scanned or imported images and exports the image and index data to more than 25 different word processing and text formats. This module was formerly known as ScanSoft OCR for InputAccel and has been updated.
- PrimeOCR Plus: Performs high-accuracy, high-reliability optical character recognition and outputs in several popular text formats. This module was formerly known as PrimeOCR and has been updated.

Indexing

The following modules capture and validate indexing data from scanned and imported documents:

- IBM CMIP-390 Index: Captures information that organizes, stores, locates, and retrieves documents from the IBM CMIP-390 system.

- IndexPlus: Web-deployable client module that transforms data from paper or electronic sources into indexed digital content. Depending on the export module in the process (IPP), the content is exported to back-end systems for fast, efficient data and image storage and retrieval. In setup mode, administrators customize IndexPlus by creating index families, index fields, image zones, keyboard layouts, keyboard shortcuts, scripts, and setting default configuration settings for optimum efficiency and productivity. To further reduce operator involvement and/or reduce keying errors, index fields can be pre-populated with information contained in external databases.

Export

The following InputAccel export modules export batch data to standardized or commonly used output systems or data formats:

- File System Export: Exports files to a specified path without altering data. During setup, you configure the export file name and location separately using a combination of hard-coded characters and IA Values. You can also configure file names, extensions, and locations in the IPP using IA Values.
- Image Export: Converts image files into specified file types, compression, and color formats. You can export a series of single page files with unique file names or a single multi-page file.
- Index Export: Converts index data stored as IA Values to ASCII text files and exports them to a local or network directory structure as ASCII text files. During setup, you create index strings for each level in the tree structure using IA Values. IA Values contain information such as the date and time images were scanned, decoded bar code values, file sizes, operator names, key entry indexing, and other values. You can customize the index strings by adding ASCII characters and applying numeric and string formats.
- ODBC Export: Enables InputAccel content to be inserted, updated, or fetched from supported databases using ODBC.
- PDF Export: Assembles PDF documents to form a single new PDF document. Documents can be written to a directory or back to the InputAccel Server.
- Values to XML: Converts IA Values into XML elements and produces XML documents. Existing XML documents can be used as templates.
- Web Services Output: Export module that enables InputAccel to use third-party services and participate in an SOA environment. It can also be used to provide a response to a web service call initiated by Web Services Input. Using the mapping information defined during setup, the module converts data represented in IA Values to a format compatible with the external Web Service.

The following modules export data directly into proprietary systems:

- ApplicationXtender Export: Enables you to connect to your ApplicationXtender 5.3 document management system. You can export and store any type of document and map IA values to index fields within document classes. This module is only installed when your system includes the ApplicationXtender software.
- Archive Export: Populates an SAP System with administrative data and content. Administrative data (indexing and other metadata) is stored on an SAP R/3 System. Captured images and other binary data is stored in a separate Content Server using the SAP HTTP Content Server Interface.
- Documentum Advanced Export: Exports documents to new or existing objects in the Documentum system. Objects are any items that can be manipulated, such as documents, document renditions, as well as the cabinets and folders in which they are stored.

- FileNet Content Manager Export: Exports InputAccel files to a FileNet Content Manager system. You can export and store any type of document and map IA values to index fields within document classes.
- FileNet Panagon IS/CS Export: Exports InputAccel content to a FileNet Panagon Image Services (IS) and Content Services (CS) server. Commits TIFF images or PDF files to the server as FileNet documents. When a document is committed, the FileNet system assigns it a document class, a unique document ID number, and index fields.
- Global 360 Export: Exports image data and index data into a Global 360 Server.
- IBM CM Advanced Export: Connect simultaneously to one or more Content Manager library servers to export InputAccel image data, non-image data, and index data by integrating with the Content Manager client software for Microsoft Windows. Through the use of item definitions, item types, export lists, attributes, routing processes, links, and references, IBM CM Advanced Export supports the entire Content Manager object model. IBM CM Advanced Export can convert images it exports into a variety of file types, color formats, and compression settings. This modules only installs when your system includes the Content Manager client software.
- IBM CMIP-390 Export: Exports image data and index data to the IBM CMIP-390 system. You can index your documents, validate the data, and then store the index data in the IBM CMIP-390 system prior to exporting the images by using the IBM CMIP-390 Index module.
- IBM CSSAP Export: Enables you to archive InputAccel content, metadata, images and index values to SAP utilizing IBM CM CommonStore. During module setup, you can choose how to process groups of pages, which items to export, where to route the items to be exported, and how to archive the items within SAP.
- MS SharePoint Export: Exports captured documents and images from the InputAccel system to the MS SharePoint Portal Server 2003 repository system.
- Open Text Livelink Advanced Export: Exports image and non-image data from InputAccel batches directly into Livelink, a client-server document storage and retrieval system. Livelink provides powerful search and document management on the Internet.

Utilities

The following utility modules perform specific tasks:

- Copy: Copies batches to another InputAccel system, to a local or network directory, or to an FTP site.
- DLL Viewer: Lists details on the DLLs in use by a running program.
- Image: Performs several image viewing alterations. Image can automatically rotate pages in 90 degree increments, generate thumbnail images, and round image width to 16-bit or 32-bit multiples. Image can process tasks independently of an operator or pause between images to enable an operator to accept each image. The module can also accept IA Values from other modules.
- Multi: Deletes pages, documents, or batches and inserts new nodes to form new documents within a batch.
- PixView: Troubleshoots problems encountered when scanning pages or processing images.
- Spawn: Launches a program depending on InputAccel task events.

- Timer: Triggers InputAccel modules to start processing tasks from a batch at a particular time.

Chapter 3

Basic Concepts

To understand how InputAccel performs information capture, you need to understand the following basic concepts:

What is a batch?

The scanning and importing modules create batches (IAB) based on a compiled process file (IAP). The IAB file contains the IAP file, which defines the modules that the InputAccel Server uses to process images and data, the order in which to use those modules, and what to do with the resulting data.

Unlike data capture systems that require the entire batch to complete processing before proceeding to the next task, InputAccel organizes a batch into as many as eight batch levels, recognized internally as levels 0-7. Batch levels, which are also known as trigger levels, represent one type of work unit. Each work unit is referred to as a task. A task includes the data to be processed, processing instructions, and identification, so that the InputAccel Server knows which batch the task belongs to when the module returns it.

The InputAccel Server sends a task to a module when the task is **Ready**, the module is available, and the batch priority is non-zero. The batch priority is a numeric value from 0 to 99 that indicates the relative urgency for processing tasks within a batch. The default batch priority is 50. Lower priority number values indicate a higher processing priority; for example, tasks from a batch with a priority of 49 would get processing precedence over tasks from a batch with a priority of 50. When the batch priority number is set to 0, the batch is considered offline and cannot issue tasks until its priority number is changed to a non-zero value.

A deactivated batch whose priority is set to 0 effectively removes it from the task processing queue. The InputAccel Server will not send tasks from a batch whose priority is set to 0. During a batch copy, batches go into an offline state to protect them from being modified. To restore an offline batch to online status (so that its tasks can be processed), change the priority setting for the batch from 0 to a positive value from 1 to 99. Tasks from batches with a lower priority number are processed before tasks from batches with a higher priority number.

Topics in this section include:

- InputAccel Values, page 20

Batch pages and files

You create batch pages when you scan or import documents into an InputAccel Server through the scanning and importing modules. A page is defined as a single-sided image, which is scanned or imported into an InputAccel Sever by modules such as Scan, ScanPlus, Multi-Directory Watch, Email Import, ECM Web Service, Web Service Input, and Ricoh GlobalScan Plug-in. For every page scanned or imported in simplex mode, one page is counted. For every page scanned or imported in duplex mode, two pages are counted. Each single-sided page is counted once when it enters an InputAccel system. Depending on the module, batch pages can be:

- Image files: BMP, TIFF, PNG, JPEG, JPEG2000, JBIG, GIF, PDA, PCX, MDA, DCX, CALS, CMP, TG4
- Non-image files: DOC, PDF, HTML, RTF, TXT, XML

Typically, these modules send a TIFF image of each document page that is stored in a stage file to the InputAccel Server that owns the batch. A batch can only exist on one InputAccel Server in a ScaleServer group.

InputAccel 6.0 introduces a new batch (IAB) naming convention and directory structure for batches and stage files. Previously batch names were based on the batch name and stage files were stored in a separate folder at the same level as the IAB file. Now, the InputAccel Server uses the batch ID to name the batch and create a directory structure for the batch and stages files. The batch ID is a 32 bit integer that consists of ten digits, which must be unique in a ScaleServer group. Using the batch ID, instead of the batch name, enables the InputAccel Server to rename batches while they are processing and removes restrictions on characters used in a batch name.

Each batch is stored in its own folder. The directory structure is based on the batch ID, which consists of ten digits that is split into three folders. For example, a batch, whose batch ID is 0123456789, is named 0123456789.iab and is stored in the Batches\0123\456\789 folder. The batch folder contains:

- **Batch file (IAB)**: A batch file contains all of the IA Values of a batch and the tree structure. As a batch proceeds through an InputAccel system, the IA Values are updated with the values generated by each module.
- **Text file**: An empty text file with a GUID (Globally Unique Identifier) name that is used for file recovery purposes.

- **Stage files**: The InputAccel Server stores images and non-image files on its hard disk as a stage file. The stage file names are based on the Node ID in hexadecimal format and the stage number. Stage file extensions are called stage numbers and correspond to the order that the modules are listed in the IPP. For example, if ScanPlus is the first module, each image file that ScanPlus sends to the InputAccel Server is identified by the ".1 " file extension. The next module in the IPP sends the InputAccel Server a stage file that is saved with a ".2" file extension. If the node ID of the page is 23e, the names of the stage files are 23e.1 and 23e.2. There can be multiple stage files per node, each representing a different "stage" or module in the IPP.

  The following table shows the record structure for Node ID 23e, which is populated when the InputAccel Server receives the stage file name stored in the `OutputImage` IA Value:

Module	Trigger IA Value	Stage file name value
ScanPlus	`OutputImage`	`<ca:9c-23e-1`

Image Enhancement	`InputImage`
	
	`OutputImage`
		`<ca:9c-23e-1`
		
		`<ca:9c-23e-2`

Image Export	`InputImage`	`<ca:9c-23e-2`

  The stage file name value `<ca:9c-23e-1` represents:

  — `<`: Designates an InputAccel stage file.

  — `ca`: Identifies the client/server communication session.

  — `9c`: Represents the Batch ID in hexadecimal format.

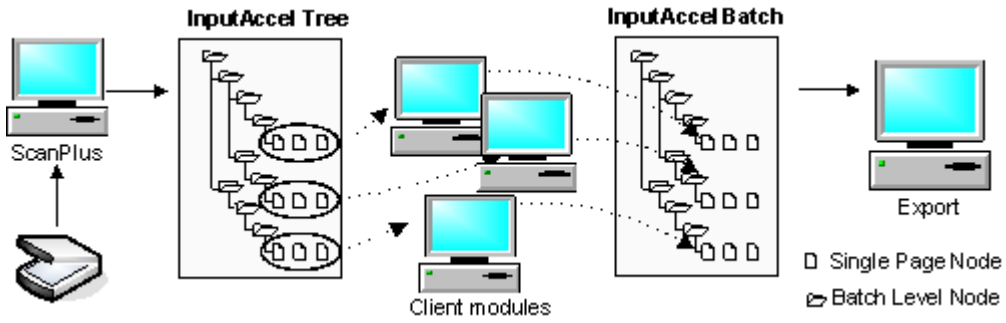
  — `23e`: Designates the Node ID in hexadecimal format that is assigned to a page node at level 0.

  — `1`: Identifies the stage file number, which represents the sequence number of the module in the process.

Tree view

The InputAccel tree is a graphical representation of the batch hierarchy. A hierarchical structure offers greater flexibility and corresponds to original filing and storage systems. When you scan or import pages into an InputAccel system, InputAccel creates a thumbnail image of each page and displays it in the tree. Each thumbnail image and folder icon is called a tree node.

Nodes are arranged in a tree structure with one "root node" and up to eight batch levels. The root node (level 7) represents the batch. Leaf nodes (level 0) represent pages. The folder icons represent higher level nodes, such as a document or folder. Within a batch, each node identifier is unique. Internally, a node consists of a record structure that stores all data associated with its corresponding batch level work unit. A node can consist of sub-nodes, IA Values, pages, and fields. The following diagram illustrates the tree and node hierarchy:



In modules, such as ScanPlus, RescanPlus, and IndexPlus, operators can organize, add, and delete nodes to maintain batch integrity. This flexibility enables operators to correct deskewed, forgotten, or misidentified pages.

InputAccel Values

InputAccel stores batch-related information in IA Values to pass data between the InputAccel Server that owns the batch and the modules defined in the IAP file. IA Values store setup and processing information that includes file and image names, module configuration settings, processing statistics, and data captured during production processing. IA Values are properties associated with a batch, process, or module. Each module has a Module Definition File (MDF) that defines the IA Values allowed for that module. The three types of IA Values are:

- **File Input/Output Variables**: Triggers stage files, which contain image and text files, to pass between the InputAccel Server and the specified module. A potential trigger value becomes an actual trigger value when it is used in the VBA code of an IPP. These files serve as pointers to stage files by storing the stage file name. While nodes can consist of work units larger than pages (such as folder nodes that consist of several pages), File Input/Output IA Values only point to page nodes. There can be multiple stage files per node, each representing a different processing stage. To pass the stage file name, the InputAccel Server sets the next input variable of the module equal to the output variable of the previous module in the IPP.

  Examples of File Input/Output IA Values are:

  — `InputImage`: Stores the names of image stage files.

  — `OutputImage`: Stores the names of image stage files.

  — `InputPDA`: Stores the names of text stage files.

  — `OutputPDA`: Stores the names of text stage files.
- **Input IA Values**: Stores default module settings that a module uses to process tasks, including default and user-defined module settings. These settings can be changed in setup mode in the Administration Console module. Most Input IA Values contain data given to a module before processing begins.
- **Output IA Values**: Output variables store data generated by the module during processing. This kind of data includes statistical information, such as the scanner operator ID, the date and time a page was scanned, and operator indexing entries. Output variables store data generated by the module during processing that can be exported.

What is a process?

You write and compile an Integrated Process Project file (IPP) in the Process Developer module using Microsoft Visual Basic for Applications (VBA). The IPP, which is a script file, contains instructions defining which modules the InputAccel Server should use to process images and data, the order in which to use those modules, and what to do with the resulting data. Compiled process files (IAP) are located in the `IAS\Process` directory. As documents are scanned or imported into an InputAccel system, a scan operator creates batches and assigns an IAP to the batch. The IAP functions as a template for the batch and is stored in the IAB file. When the InputAccel Server starts processing a batch, the modules store all batch-related information in the IAB file, without affecting the information of the original IAP file.

The IPP ties the InputAccel Server and client modules together. Processing instructions include:

- Job identification
- Module identification
- Module processing order
- Module processing instructions
- Data processing instructions

The InputAccel installation includes sample processes that include many of the modules and demonstrate programming principles. For a complete description of IPP functionality, refer to the Process Developer Guide.

Topics in this section include:

Steps

A step refers to a specific configuration of an InputAccel module in an Integrated Process Project file (IPP). In an IPP, you can use the same module in different steps to perform different tasks.

For example, to highlight information that you want indexed on a page, create two IndexPlus steps with different names in an IPP. In the first step, operators rubber band an index zone with a mouse to highlight data on an image; in the second step, operators type the highlighted information into an index field. Using multiple steps of the same module in an IPP increases productivity by isolating types of work and operators increase efficiency by performing a single type of task in each step.

Module Definition Files

A Module Definition File (MDF) is a text file that defines the IA Values variables that are allowed for that module. InputAccel stores data in the defined variables (and custom variables you add to an MDF) and passes the data between modules. Use the Process Developer module to specify an MDF for each module step included in an IPP.

MDFs for all of the InputAccel modules are provided in the `InputAccel\Client\src\ipp` folder and can be edited with a text editor.

Trigger levels

You select a trigger level when you add a module step to your IPP. The trigger level controls the size of the task that the step receives during production. A level 0 task contains the data from a single page; a level 1 task contains the data from a document, which may hold several pages; a level 7 task contains the data from all of the pages in an entire batch. InputAccel maintains this hierarchy throughout processing, which enables captured data to export as one unit with structural divisions.

For example, if a batch includes pages from a storage facility, classify the pages into the following work unit types: documents, folders, filing cabinet drawers, storage boxes, rows, and warehouses. The familiar work units reduce operator confusion and provide an easy reference system:

- Level 0 = Page
- Level 1 = Document
- Level 2 = Folder
- Level 3 = Filing cabinet drawers (User defined.)
- Level 4 = Storage boxes (User defined.)
- Level 5 = Rows (User defined.)
- Level 6 = Warehouse (User defined.)
- Level 7 = Batch

Trigger levels also control locking. When a task is triggered at level 1, then it is held (locked) by the module until the entire level 1 task node has been completely processed. Locking maintains batch integrity by preventing access to a task by more than one user.

The following process example explains how trigger levels work:

- **Step 1**: The ScanPlus module inputs data into the system, and then the InputAccel Server assumes control.
- **Step 2**: Image Enhancement is triggered at level 0 (page level). This step receives one page at a time as it becomes available from the InputAccel Server. The page is returned to the InputAccel Server after it is processed.
- **Step 3**: NuanceOCR is triggered at level 1 (document level). If there is a document of 50 pages and Image Enhancement only released 49 pages to the InputAccel Server, then NuanceOCR cannot begin processing. The InputAccel Server must receive all 50 pages from Image Enhancement before it can send the task to NuanceOCR.
- **Step 4**: Image Export is triggered at level 7 (batch level). All previous modules must complete processing the batch before Image Export receives its task and begins exporting.

Refer to the Process Developer Guide for additional information regarding trigger levels.

Note: Trigger levels are different than trigger variables, which are File Input and Output IA Values that notify the InputAccel Server when a task is ready to be processed by the next step. Trigger variables enable the InputAccel Server and client modules to work on a "push" basis, by pushing a task through the system as soon as it is ready for the next stage in processing.

Departments

Departments in InputAccel are an optional feature that enable the classification and routing of work to specific workstations or users that specialize in processing certain types of tasks. Departments are associated with an IPP step and are recognized by the InputAccel Server where the IPP is installed. The department designation modifies the step and instructs the InputAccel Server to send tasks from that step only to workstations that are connected as members of that department.

Departments are defined by an administrator in the Administration Console. Each department is assigned an access control list (ACL), based on role permissions set for users and groups. These definitions secure access to departments and ensure that tasks are sent to the appropriate workstations for a department. During production, operators can log in as members of a department to receive and process work for that department.

Refer to the Using Administration Console section of the Administration Guide for additional information on creating departments. Refer to the InputAccel Documentation for information on configuring departments for each module.

Event handlers

Using the Process Developer module, you program each step in the IPP with Microsoft Visual Basic for Applications (VBA). Typically, an IPP program contains event handlers, which are units of code that perform a task or tasks, but does not return a result. The standard event handlers are:

- `Finish`: Contains processing instructions that are executed by the InputAccel Server immediately after a module successfully completes a task. For most module steps, you must use a `Finish` event handler to tell the InputAccel Server where to send the stage file(s) and other task related data generated by a module.
- `Error`: Contains processing instructions that are executed by the InputAccel Server when a client module fails to successfully process a task.
- `Prepare`: Contains processing instructions that are executed by the InputAccel Server before sending a task to a module step. `Prepare` event handlers are commonly used to initialize values in the step.

Refer to the Process Developer Guide for additional information on creating event handlers.

Topics in this section include:

- Basic routing in Finish event handlers, page 24
- Conditional routing in Finish event handlers, page 24

Basic routing in Finish event handlers

Use `Finish` event handlers for each step in the IPP to control data flow in an InputAccel system. Within a `Finish` event handler, you set the output of one step, such as data, images, or files, as the input of the next step. Typically, export steps do not require `Finish` event handlers because data routes to a back-end system rather than an InputAccel module.

To route data in a `Finish` event handler from one module step to another, assign the Output IA Value of the first module to the Input IA Value of the next module. For example:

```
p.step2.InputIAValue = p.step1.OutputIAValue
```

The following `Finish` event handler for ScanPlus routes a File Output IA Value (`OutputImage`) to the file Input IA Value (`InputImage`) of an NuanceOCR module. File Input/Output IA Values are the trigger variables that notify the InputAccel Server that a task is ready to proceed to the next step.

```
 Private Sub scan_Finish(ByVal p As IASLib.IAS_RECORD_0)
 p.nuanceocr.InputImage = p.scanplus.OutputImage
End Sub
```

Conditional routing in Finish event handlers

Add code in the `Finish` event handler to route data to different modules based on the output of the previous module by using a standard Visual Basic `If...Then...Else` statement.

For example, the following `Finish` event handler for the Image Quality Assurance step (QAImage) includes two processing options:

- When image processing is unsuccessful, the image routes to the RescanPlus (RescanPlus.InputImage) step for reprocessing with a message (RescanPlus.RescanReason) stating “Wrong image”.
- When image processing is successful, the image routes to the Image Export (ExportImage) step for export to a back-end system.

```
Private Sub QAImage_Finish(ByVal p As IASLib.IAS_RECORD_0)
     If p.QAImage.WrongImage = -1 Then
      p.RescanPlus.InputImage = p.QAImage.OutputImage
      p.RescanPlus.RescanReason = "Wrong image"
     Else
      p.ExportImage.InputImage = p.QAImage.OutputImage
    End Sub
```

Client-side scripting

A client-side script is a program that runs as part of an InputAccel process. You can create script actions and associate them with specific events that are defined in each InputAccel module. When the event occurs, your script action is executed.

For example, when you want to limit the number of pages within each node of a document, you create an event handler that displays a warning if more than five pages exist within a single node. If

this event handler is assigned to the ScanPlus module event named `AfterPageAdded`, a warning displays when a ScanPlus user adds a sixth node.

Scripts are written as Microsoft .NET assemblies, typically using either VB.NET (Visual Basic) or C# as the programming language. Create an assembly externally and import the resulting DLL file to a process or a batch, or use the built-in script editor and compiler to create assemblies directly within the client module Setup interfaces.

Scripts are run directly by client modules. Many modules have scripting interfaces that a script can implement in order to automate tasks associated with the module. For example, to automate OCR, create a script that uses either a NuanceOCR interface or a PrimeOCR Plus interface, then open the appropriate module in setup mode and configure it to run the script.

Compiling a process

After you finish creating an IPP, compile it using Process Developer and then install it on the InputAccel Server using Process Developer or the Administration Console module. After it is installed, use the Administration Console module to run each module step in setup mode and configure the default settings. You can set up a module for production mode at two levels:

- **Process level**: Configures the default settings for a module. The settings are saved to the process step defined in the IPP. All future batches based on the process step inherit its settings.
- **Batch level**: Overrides the default process settings of a module for a single batch. The modified settings are saved to the batch step. Future batches created with the process do not inherit the modified settings.

Chapter 4

Understanding InputAccel in production mode

In production mode, the scanning and importing modules create batches (IAB) based on a compiled process file (IAP). The IAB file contains the IAP file, which defines the modules that the InputAccel Server uses to process images and data, the order in which to use those modules, and what to do with the resulting data.

Unlike data capture systems that require the entire batch to complete processing before proceeding to the next task, InputAccel organizes a batch into as many as eight batch levels, recognized internally as levels 0-7. Batch levels, which are also known as trigger levels, represent one type of work unit. Separating batches into smaller, user-defined work units that can move through the system independently, enables InputAccel to increase throughput. Each work unit is referred to as a task. A task includes the data to be processed, processing instructions, and identification, so that the InputAccel Server knows which batch the task belongs to when the module returns it. Tasks are associated with a node and step of a batch. When the client module completes a task, it returns the task to the server. Then, the InputAccel Server queues the task for the next available processing module defined in the IAP. If multiple client workstations are running the same module, the InputAccel Server sends the tasks to the first available workstation. A task is locked when it is being processed and is unavailable to other workstations.

This interaction between the InputAccel Server and the modules enables asynchronous processing of tasks. Because each task is self-contained the modules can process tasks from any batch without regard to which pages go with which document capture job. The InputAccel Server keeps track of the original page hierarchy defined within each page's respective batch.

A task status indicates the current processing state of the batch. Valid task status values are:

- **Done**: The module step has finished processing the task.
- **Not Ready**: No tasks are currently queued for the module step.
- **Ready**: Tasks are queued for the module step.
- **Working**: The task is currently being processed by the module step.
- **Hold**: Tasks associated with the module step are on hold.
- **Error**: Tasks associated with the module step are flagged for error.
- **Sent**: Tasks associated with the module step are sent by the InputAccel Server.

Only **Hold** and **Error** statuses can be manually set for a batch within the Administration Console; all other statuses listed here are displayed by the InputAccel system during processing and cannot be manually modified. Changing the batch status to **Hold** or **Error** does not affect the batch priority.

Topics in this section include:

Understanding production modes

In production mode, the InputAccel Server processes batches based on instructions in the compiled process file (IAP). Click on the desktop shortcut for the client module to log in and begin batch processing. After successfully starting the module, the InputAccel Server connects and the client module production mode window displays.

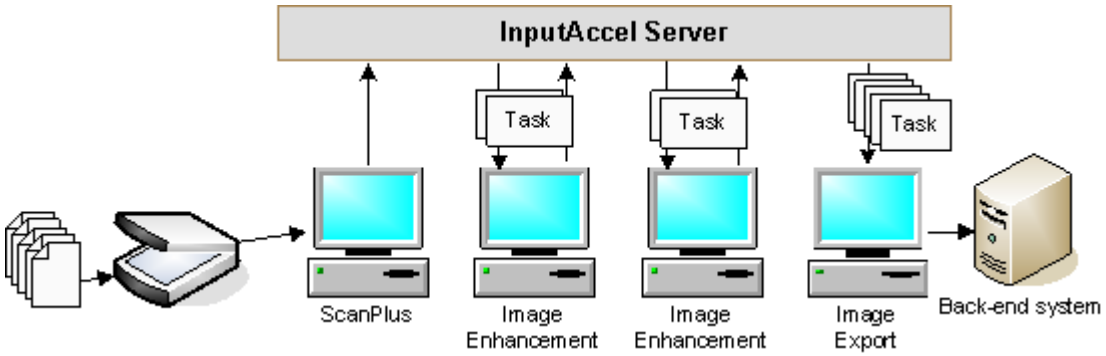
InputAccel offers two processing modes:

- **Wait for Task**: Typically, client modules run in Wait for Task mode and wait for tasks from the InputAccel Server. InputAccel has two Wait for Task modes:
  - **Run All Batches**: The module receives any queued tasks from any open batch as the tasks become available from any connected InputAccel Server. If multiple workstations are running the same module, then the InputAccel Server sends tasks to each workstation.
  - **Run Single Batch**: The module receives tasks from a single batch selected from a connected InputAccel Server.
- **Open Batch**: Some of the modules, such as File System Export and PDF Export, allow processing in **Open Batch** mode, which is typically used when testing a new process. This mode enables batch processing at any time, regardless of whether the batch is queued on an InputAccel Server for the module or includes the module in its process. The InputAccel Server that owns the batch ignores the standard `Prepare` and `Finish` event handlers in the IPP and processes the selected batch. When the module finishes processing all the tasks from the batch, the operator closes the batch and then reopens it at a workstation running the next module used in the IPP.

How InputAccel processes batches

In this example, the InputAccel Server controls batch processing after receiving data from the ScanPlus module and routes tasks to available modules based on the instructions contained in the batch file (IAB).

The following diagram illustrates the interaction between the InputAccel Server and client modules:



The InputAccel Server monitors all the client workstations and sends them tasks from any open batch. Most modules prefetch three tasks from the InputAccel Server, storing two tasks locally and working on one at all times. Each server does its own task scheduling without coordinating with other servers. The server processes batches that have the same priority based on the creation date and time. This allows the task scheduler to give preference based on the creation date and time when batches are copied from one server to another. Workstations can be configured to receive work from multiple InputAccel Servers, which increases throughput.

Modules return finished tasks to the InputAccel Server and start processing the next task stored locally. When the InputAccel Server receives the finished task, it sends the module a new task and sends the finished task to the next module in the process. If none of the workstations are available to process the task, then the InputAccel Server queues the task. This exchange is made possible by trigger IA Values, which signal InputAccel Servers to send a task to a module for processing. The InputAccel Server and modules work on a "push" basis. The push of data takes place when the name of a stage file stored in an Input or Output IA Value is passed to the next module.

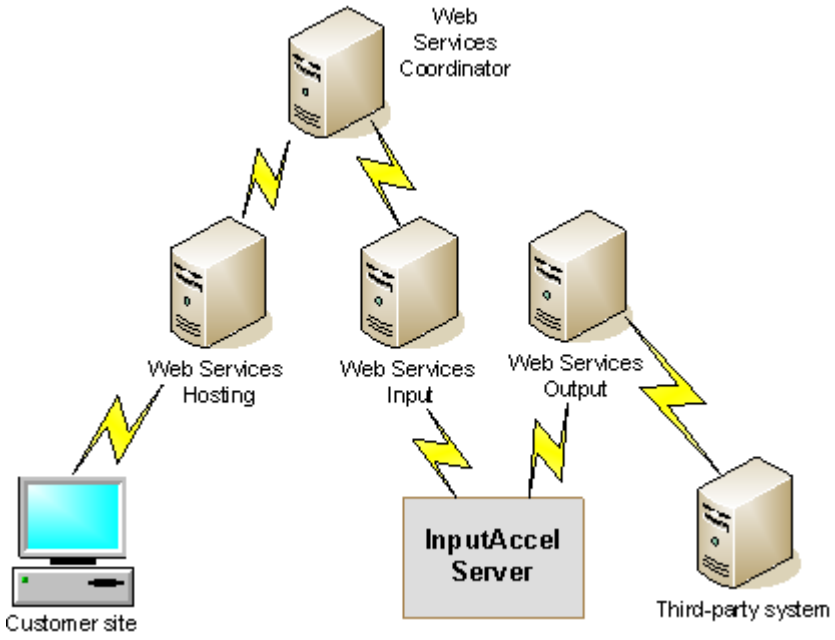
As the InputAccel Server pushes tasks through the system, the nodes in the tree hierarchy are updated with IA Values. Within a batch, each node identifier is unique. Internally, a node consists of a record structure that stores all data associated with its corresponding batch level work unit, which includes sub-nodes, IA Values, pages, and fields.

Related Topics —

What is a batch?, page 17
What is a process?, page 21
Chapter 9, Sample processes

How InputAccel processes batches using Web Services

InputAccel uses Web Services provide the standards and tools to build an SOA to process work over the Internet and enable the InputAccel system to be a consumer or provider of web services. A Web Service is a self-describing, self-contained, modular software application that provides one or more business functions to other systems through a standard Internet connection. In the following example, the customer site, the InputAccel Server, and the third-party export system are situated in different locations:



The components in this example include:

- Customer site: Sends batch information with attached documents or a URL that references a document to the Web Services Hostings service based on a connection that is defined in the Administration Console module. Web Services Input downloads the document file referenced by the URL. The documents are sent to the InputAccel Server as a stage file according to the parameters mapped in the Web Services Input module.
- Web Services: The following transactions occur:
  - The Web Services Hosting component acts as a Web Services Server and receives requests.
  - The Web Services Coordinator component manages Web Services Hostings. It also maintains the request/response data in the InputAccel SQL database and provides routing between Web Services Hosting and Web Services Input.
  - The Web Services Input module receives documents or downloads the document file referenced by a URL. It extracts input IA Values and forwards them to the Web Services Coordinator component and then waits for output IA Values to send to the InputAccel Server. In Web Services Input setup mode, administrators map web service call parameters to IA Values to determine what information from incoming parameters should be stored in IA Values and what information should be sent to outgoing parameters from IA Values. The Web Services Input module creates a batch and sends the IA Values and stage files, if any, to the InputAccel Server.
- InputAccel Server: Runs a process that includes the IndexPlus module, which indexes and validates data sent to it by the InputAccel Server. When IndexPlus finishes indexing and validating, it returns the batch to the InputAccel Server, which updates the IA Values and creates a new stage file for this step.
- Third-party system: The InputAccel Server pushes the batch to the Web Services Output module, which routes the verified data to a third-party system over the Internet.

Refer to the Web Services Guide for more information on the Web Services architecture.

Using alerts to monitor remote client processes

EMC Captiva Input Management Console monitors the performance of InputAccel components and sends proactive alerts to administrators when problems occur in an InputAccel production environment. Refer to the Installation Guide for information on installing the EMC Captiva Input Management Console application.

Chapter 5

Securing an InputAccel system

In an InputAccel system, administrators authenticate users using the security features of Windows operating systems. After a user is authenticated, administrators use the Administration Console module to configure what a user is allowed to do in an InputAccel system. Roles and the permissions assigned to every role are stored in the InputAccel Database. The security controls are:

- **Roles**: Defines roles allowed for InputAccel users. The roles define the actions allowed for users by assigning members and their permissions to a role. Roles are viewed, added, or modified from within the **Security** pane. By default, the Administration Console grants all permissions to administrators on the machine where the Administration Console is running. Administrators can configure roles and assign permissions to individuals responsible for working in an InputAccel system. The permissions are determined by functions for a specific module.
- **Access Control List (ACL)**: Assigns user or group permissions for modules, batches, departments, or processes. From an Administration Console pane that displays modules, batches, departments, or processes, administrators select an object, right-click and select **View Selected > ACLs**. The **Access Control List** window displays the list of available ACL permissions and allows administrators to select object-specific permissions for a user or group.

Refer to the Using Administration Console section in the Administration Guide for additional information on configuring security settings in an InputAccel system.

Chapter 6

Understanding the InputAccel Database

InputAccel 6.0 stores configuration settings and processing information generated by client modules in the InputAccel Database. By using a central location to store a subset of batch data, security and configuration settings, and logging information, administrators can efficiently manage an InputAccel system online from the Administration Console. A single InputAccel Database and Administration Console server can handle multiple InputAccel Servers, whether or not they are configured in a ScaleServer group.

As InputAccel processes batches, the InputAccel Database collects a subset of batch data and processing information that administrators use to manage, tune, and control the system. The InputAccel Database is a required component that does not require an InputAccel license because it is a data repository rather than an executable. The InputAccel Database stores the following information:

- Configuration settings that are needed by InputAccel components. Administrators can modify settings without interrupting or impacting the InputAccel Servers.
- License codes.
- Logging rules that are used to capture errors, audit data, and other values for use in various displays and reports.
- Data on work-in-progress, enabling requests for batch information to be fulfilled without requiring the InputAccel Servers to open every batch. This dramatically improves InputAccel Server performance in situations where administrators need to view batch data.
- Settings for batches, tasks, and batch IA Values.
- Web Services subsystem configuration.

Refer to the Administration Guide for additional information on the InputAccel Database.

Chapter 7

Controlling image formats

InputAccel enables administrators to configure image processing modules for optimum performance. In particular, the color compression setting choices configured in setup mode for modules such as ScanPlus, RescanPlus, and Image Enhancement, affect the image quality, file size, processing speed, and module performance. For example:

- The JPEG color compression format produces smaller image sizes efficiently, but reduces image quality. If the same JPEG compressed image is modified by several InputAccel modules, then the compounded compression losses may not be acceptable.
- The Packbits compression format preserves image quality and uses little or no processor bandwidth, but produces large files.
- The ZIP and JPEG 2000 color compression formats produce good quality images, but uses significantly more CPU bandwidth and decreases module efficiency.

Refer to the guides in the InputAccel Documentation that document scanning and image processing for more information on configuring image formats in setup mode to meet business requirements for image quality, processing speed, and image file size.

Chapter 8

How information lives

InputAccel enables organizations to capture information from virtually any paper or electronic source and transform it into usable, business ready content, and deliver it into a variety of back-end systems. In the following example, InputAccel enables a bank to reduce costs by processing a loan application globally over the Internet:

Starting the data capture process:

1. A customer completes a loan application at a local bank branch office in Stuart, FL.
2. A loan officer approves the application and faxes a copy to the credit department in the main office, which is located in Toledo, OH.
3. A bank employee initiates an InputAccel process for loan applications over the Internet and attaches the URL of the faxed image. The compiled process file (IAP) defines the steps used to capture the data in the faxed image and instructs the InputAccel Server on how to process the batch. The process includes the Web Services Input, NuanceOCR, IndexPlus, and Web Services Output modules.
4. The first step in the process is the Web Services Input module, which creates a stage file and a batch file (IAB) based on the mappings of parameters to IA Values. The stage file contains the faxed image file that was downloaded from the URL. For security, the bank uses HTTPS (Hyper Text Transfer Protocol Secure) for all Internet transactions.

Capturing data:

1. The Web Services Input module forwards the batch and stage file to the InputAccel Server, which is located in Mexico City, Mexico and uses the mapping information defined during Web Services Input setup to convert data represented in IA Values to an external Web Service using the client side scripting and/or mapping configuration. To reduce expenses, the bank uses the processing center in Mexico City to perform the recognition and indexing steps.
2. The server pushes the task to the NuanceOCR module. The loan application is a structured document that contains data in the same area of every page. In NuanceOCR setup mode, an administrator defines zones to capture data using zonal OCR in production mode.
3. NuanceOCR recognition processing automatically captures the customer's name, address, and social security number and stores the data in IA Values. The module passes the IA Values to the InputAccel Server and the data is added to the batch file (IAB).
4. The InputAccel Server pushes the task to IndexPlus for an operator to verify the data. An InputAccel administrator defines index fields for the captured data in IndexPlus setup mode and writes scripts that verify the address against a United States Postal Zip Codes database. In

production mode, the operator compares the data to the image and verifies that the captured data is accurate. When the operator clicks the **Accept Task** link, the task, including the data stored in IA Values, proceeds to the InputAccel Server.

5. The InputAccel Server updates the IA Values and sends the extracted social security number to the Web Services Output module.
6. The Web Services Output module issues a web service call to a credit bureau requesting the customer's credit history based on his social security number.
7. The Web Services Input module receives the customer's credit history and passes the information to the InputAccel Server.

Exporting data:

1. The InputAccel Server pushes the task to the Web Services Output module.
2. The module exports the customers credit card history and indexed fields to a back-end repository system located in Toledo, OH.

Using data:

1. A loan officer reviews the customer's credit history and approves the loan. The data remains on the back-end system for future use.
2. The bank credits the customer's account at his local bank in Stuart, FL.
3. A satisfied customer buys a boat.

Chapter 9

Sample processes

InputAccel installs the sample Integrated Process Project (IPP) files for demonstration purposes. When creating a custom IPP, consider any special dependencies between modules or between the InputAccel Server and a third-party system. For example, a process that contains a RescanPlus step must also contain a ScanPlus step for RescanPlus to work properly. Likewise, some export modules require the installation of a third-party back-end or client software to run.

For help creating or modifying processes, refer to the Process Developer Guide. To learn more about installing and monitoring processes, refer to the Using Administration Console section in the Administration Guide.

The following table lists the module and associated sample IPP file(s):

Module	**IPP file**
ApplicationXtender Export	Export21
Archive Export	Export20
Auto Annotate	Annotate1
Copy	Iacopy1
ECM Web Services Importer	DocumentumRescan
Email Import	EmailImport1
Eastman Software Imaging Server Compatible Export	Export3
Documentum Advanced Export	• _DocumentumCapture_52_Sample.ipp (Advanced) • DocumentumRescan • Export5
File System Export	• EmailImport1 • Export19 • PageReg
FileNet Content Manager Export	Export22
FileNet Panagon IS/CS Export	Export4

Module	IPP file
Global 360 Export	Export2
IBM CSSAP Export	Export16
IBM CM Advanced Export	Export14
IBM CM Advanced Export	Export7
IBM CMIP-390 Export	Export7
Multi	All
Image Divider	Iamdw
Image Enhancement	• _DocumentumCapture_52_Sample.ipp (Advanced) • Capture3, 7 • Image2, 4 • Index3, 6, 7
Image Export	• Annotate1 • EmailImport1 • Iacopy1 • Iamdw • Image1, 2, 4, 5, 6 • Index1, 3, 5, 6, 7, 9, 10 • Objects1 (Advanced) • Rescan7 (Advanced) • Scan7 (Advanced) • XMLobj (Advanced)
Image Quality Assurance	• Image4, 6 • Index3, 4, 6, 7, 9
Index Export	• Index1, 3, 5 , 6, 7, 10 • IndexNotify • ManualRetrigger • Objects1

Module	IPP file
IndexPlus	• _DocumentumCapture_52_Sample.ipp • Export2, 4, 5, 8, 9, 14, 16, 18, 20, 21, 22 • Image5 • Index10 • IndexNotify • ManualRetrigger • Objects1 • Rescan7 • Scan7 • XMLobj
MS SharePoint Export	Export18
Multi-Directory Watch	Iamdw
NuanceOCR	• _DocumentumCapture_52_Sample.ipp (Advanced) • Capture7 • PageReg
ODBC Export	Export8
Open Text Livelink Advanced Export	Export9
PDF Export	Capture3, 7
Page Registration	• _DocumentumCapture_52_Sample.ipp (Advanced) • PageReg
PrimeOCR Plus	Capture3
RescanPlus	• _DocumentumCapture_52_Sample.ipp (Advanced) • DocumentumRescan • Export7 • Image4 ,6 • Index5, 6, 7 • PageReg • Rescan7

Module	IPP file
ScanPlus	All
Values to XML	• Image5 • XMLobj (Advanced)

Understanding the advanced processes

The advanced sample IPPs are located in the `InputAccel\Client\src\ipp\advanced` directory and provide process developers with examples of how to use certain methods. The sample IPPs are provided for demonstrations purposes only. For information on creating and using processes, refer to the Process Developer Guide. The advanced sample IPPs are:

- `_DocumentumCapture_52_Sample.ipp`: Configures each module step before creating any batches. Uses the Process Developer module to view the comments describing each step in the process. This sample IPP uses the following modules:
  - ScanPlus
  - Image Enhancement
  - Page Registration
  - Multi (Split Documents)
  - NuanceOCR (OCR for Indexing)
  - IndexPlus
  - RescanPlus
  - NuanceOCR (Create Documents)
  - Multi (Ready to Export)
  - Documentum Export (exporting images, PDF, and text)
  - Custom (Data)
  - IADone
  - Multi (DeleteBatch)
- `BatchProcessRoutines.ipp`: Demonstrates the following routines:
  - Batch_Create: Sets initial values for the various levels. For example, set a user-defined per-batch value that provides information about who created the batch.
  - Batch_Delete: Logs information about who deleted the batch to the NT event log. Set the InputAccel Server to log events to the NT event log. Do not trigger module steps to run with this routine because the batch will be deleted before the steps have a chance to execute.
  - Process_Install: Sets a user-defined per batch value that provides information about who created the process.

- `Collections.ipp`: Demonstrates how to iterate through an `IAValuesCollection`. The collection is defined in `collections.mdf`. The advantage of using an `IIAValuesCollection` is the ability to use the `For each` statement to iterate the collection thus making the code look more intuitive. For example:

```
Private Function GetInvoiceTotal(collection As IIAValuesCollection)
    Dim FieldDetail As IAValuesIterator
    Dim lTotal As Long

    For Each FieldDetail In collection
        If (FieldDetail.Name = "Total") Then
            If (FieldDetail.Value.Flagged = False) Then
                'only add the total if it is not flagged
                lTotal = lTotal + CLng(FieldDetail.Value.Data)
            End If
        End If
    Next FieldDetail

    GetInvoiceTotal = lTotal

End Function
```

- `Globals.ipp`: Demonstrates the use of Objects, IAVariants, and the activation of the Jobs combo box. This sample uses two custom MDFs, both of which define the same Global variable and IAVariant but follows the rule of multiple definitions for Globals and IAVariants. In addition, the sample demonstrates how to activate a new combo box by defining and initializing the following Global IA Values:

  — `IABatchComboChoices`

  — `IABatchComboTitle`

  — `IABatchComboResult`

  When this process is used to create a batch, the user will see an additional list box in ScanPlus. By selecting different items for each batch, notice that the values assigned to the Client object are changed after a batch finishes.

- `IndexNotify.ipp`: Provides examples of how to use the `Notify` routine. This routine is called by the IndexPlus module when a non-task node has been modified. The sample ensures that Index Export is retriggered so it can export the updated value.

  Note: If the Index Export step has not yet been triggered, then setting the `Ready` trigger alone will not trigger it to run; the `Index_Finish` routine must set the `InputImage` values as well. In order to see this functionality, set up the IndexPlus module such that the document opens at a specified level above the task level, and confirm that the editing of non-task nodes is allowed.

- `ManualRetrigger.ipp`: Provides an example of manual triggering and also provides five examples that illustrate moving and deleting nodes. These examples assume that the IA tree is not visible above level 2, so there is no need to handle moves or deletes above that level.

- `Objects1.ipp`: Demonstrates the use of Objects, IaVariants and Any (a variable). To use the sample, compile the IPP into an IAP and then install the process on the server. Using the Administration Console module, load settings from the file `objects.txt` (in `c:\Program Files\InputAccel\Client\src\ipp\advanced`). Scan the images installed in `c:\Program Files\InputAccel\Client\images\object example images`. Index some of the data and then run two steps of Index Export, using one with “Int” department and view the resulting text file.

- `Rescan7.ipp`: Provides an example of how to trigger the RescanPlus module at level 7 and set trigger values in the `Error` routine so that the task is queued correctly if the batch is closed or terminated before completion. The example implements triggering new nodes added during rescan by defining two routines in the Main module: `SetNewNode` and `GetNewNode`. These routines set and get a dynamic IA Value that is used as a flag for new nodes. The routine then flags each new level 0 node (page) using `SetNewNode` in the `Tree_PostNodeAdd0` routine. The IPP then clears the new node flag in the `Rescan Error` routine so that the RescanPlus module knows what new nodes have been created. Finally, it checks the new node flag in the `Rescan Error` routine. If the new node flag is true, then the `Error` routine knows there is a new node and can set `InputImage = OutputImage`, thereby setting the trigger. If you define any other triggers in the Rescan MDF and subsequently use those triggers in the IPP, then you also need to set those triggers in the `Error` routine.
- `Scan7.ipp`: Provides an example of how to trigger the ScanPlus module at level 7.
- `TreeFunctions.ipp`: Provides an example of how to log events to track the order in which they are called. Each step Object in this IPP (Batch, Process, Tree, and Scan) writes to a scan keyentry value. These values can be seen in the ScanPlus module, or by using the Administration Console under the tree values for that batch. Refer to the Using the Administration Console section of the Administrator Guide for information about viewing or modifying batch IA Values.
- `XMLObj.ipp`: Illustrates the use of an object to more efficiently move data to and from the InputAccel Server. In this example, a few values have been mapped to an object that is declared in its own MDF. That object is then transferred to the Values to XML module, which extracts the values from the object and performs its usual function of mapping them to an XML step document based on a predefined template. The performance difference when transferring a single object instead of eight IA Values is negligible. Also, the use of objects is not limited to the Values to XML module. Many customers use various InputAccel export modules to export hundreds or thousands of values per node, each requiring at least one network request between the InputAccel Server and InputAccel Client. By transferring all of these values in a single object, replace all of these individual network access requests with a single request to transfer the object. The resulting performance improvement can be dramatic. This sample IPP works with two other provided files:
  - `xmlobj.txt`: This is the step setup data for the included IPP, and is located in the same folder as the IPP. After compiling the IPP into an IAP and installing it on the InputAccel Server, use the Administration Console module to view the process and load the settings contained in this file into the process.
  - `xmlobj.mdf`: This MDF is located in the `InputAccel\Client\src\ipp` folder and must be included in the process. It declares the object for mapping IA Values.

Understanding the sample processes

InputAccel installs the sample Integrated Process Project (IPP) files in the `InputAccel\Client\src\ipp` directory and the compiled IAP files in the `\IAS\process` directory for demonstration purposes only. To change the underlying logic behind any of these processes, open, modify, and recompile the IPP using the Process Developer module, and then install the new process on the InputAccel Server. The sample processes are not configured; setup each module step through the Administration Console module before using the process in a production environment. For information about modifying the underlying IPP code used in one of these processes, or creating a new IPP, refer to the Process Developer Guide. For information

about configuring the individual module steps within a process after it has been compiled, refer to the individual user guide for each module. The following list includes the sample IPPs and a short description of their steps:

- `Annotate1.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Stamp: Adds annotations and merges them with the images using the Auto Annotate module.
  - ImageEx: Exports annotated images to a back-end system using the Image Export module.
  - DeleteBatch: Deletes the batch when it finishes processing using the IAMuli module.
- `Capture3.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - IE: Applies image enhancing filters for improved image quality using the Image Enhancement module.
  - PrimeOCR Plus: Converts images to PDF files or other text and word processing files.
  - PDFExport: Concatenates PDFs into multi-page files and exports to a back-end system using the PDF Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Capture7.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - ImageEx: Exports annotated images to a back-end system using the Image Export module.
  - SSOCR: Converts images to PDF files or other text and word processing files using the NuanceOCR module.
  - PDFExport: Concatenates PDFs into the multi-page files and exports to a back-end system using the PDF Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using a virtual module (IADone).
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `DocumentumRescan.ipp`:
  - RescanImp: Uses the ECM Web Services Importer module to process “rescanned” batches received from the EMC Documentum Server.
  - Scan: Although included in the process, this step is not used.
  - Rescan: Used to view thumbnail images.
  - DMExport: Exports data using the Documentum Advanced Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.

- `EmailImport1.ipp`
  - ImportEmail: Imports document pages into the InputAccel Server using the Email Import module.
  - ExpImg: Exports annotated images to a back-end system using the Image Export module.
  - ExpFile: Exports annotated images to a back-end system using the File Export module.
  - Hold: Holds batches using the Multi module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Export2.ipp`
  - Scan: Scans or imports document pages into InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - ExpWangNT: Exports images and index information using Global 360 Export to a Global 360 Export Server.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Export4.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - ExpFNet: Exports images and index information to FileNet Panagon IS/CS Export.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Export5.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - ExpDoc: Exports images and index information to Documentum Advanced Export.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.

- `Export7.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index_NoHost: Indexes and rejects images with poor quality using the IBM CMIP-390 Export module.
  - Rescan: Rescans rejected images using the RescanPlus module.
  - Index_With_Host: Validates indexing data using the IBM CMIP-390 Export module.
  - ExpFAF: Export images and index information to IBM CMIP-390 Export.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Export8.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - OdbcExp: Export images and index information to an ODBC-compliant database using the ODBC Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Export9.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - LiveLinkExp: Exports images and index information using Open Text Livelink Advanced Export
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Export14.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - ExpCM: Exports data to IBM CM Advanced Export.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.

- `Export16.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - ExCSSAP: Exports data to an IBM CSSAP Export system.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Export18.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - Shrpnt2003: Exports data to an MS SharePoint Portal Server 2003 server.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Export19.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Export: Exports a specified stage file to a specified file system path without changing the data in any way using the File Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Export20.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - ExSAPAL: Exports the images (content) to a content server using the SAP HTTP Content Server Interface and indexes data (and other IA Values) to an SAP system through the SAP ArchiveLink Interface using the Archive Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Export21.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - EXAX: Exports the images and index data to specified applications using the ApplicationXtender Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.

- `Export22.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - ExFNCM: Exports the images and index data to the FileNet Content Manager object store using the FileNet Content Manager Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Iacopy.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Copy: Copies the batch to a second InputAccel Server using FTP (or to a zip file for later transfer) using the Copy utility.
  - ImageExp: The module is connected to the second server. Exports the images to a local or network directory. Image Export requires two triggers before it can begin processing. The first is set by the `Scan_Finish` event handler and the second is set by the `Copy_Finish` event handler.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `IAMDW.ipp`
  - Import: Imports document pages into the InputAccel using the Multi-Directory Watch module
  - Hold: Holds the batch using the Multi module.
  - ImageDivider: Divides multi-age TIFF images using the Image Divider module.
  - Index: Indexes images using the IndexPlus module.
  - ExportImage: Exports images using the Image Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Image1.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - ImageExp: Exports images to a back-end system using the Image Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.

- `Image2.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - IE: Applies image enhancing filters for improved image quality using the Image Enhancement module.
  - ImageExp: Exports images to a back-end system using the Image Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Image4.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - IE: Applies image enhancing filters for improved image quality using the Image Enhancement module.
  - QA: Performs quality assurance using the Image Quality Assurance module. Checks image quality. Rejected images are routed to RescanPlus. Accepted images are routed to Image Export.
  - Rescan: Rescans images using the RescanPlus module.
  - ImageExp: Exports images to a back-end system using the Image Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Image5.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - Index: Indexes images using the IndexPlus module.
  - ExImg: Exports images to a back-end system using the Image Export module.
  - XML: Converts IA Values to XML data and export to a back-end system using the Values to XML module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.

- `Image6.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - QA: Checks the image quality of every fifth image using Image Quality Assurance. Rejected images route to RescanPlus. Accepted images route to Image Export.
  - Rescan: Rescans rejected images using the RescanPlus module. After the rejected image is rescanned, it is routed to Image Quality Assurance. If an operator does not rescan the image, it automatically proceeds to Image Export.
  - ImageExp: Exports images to a back-end system using the Image Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Index1.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - ImageExp: Exports images to a back-end system using the Image Export module
  - IndexExp: Exports index data to a back-end system using the Index Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Index3.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - IE: Applies image enhancing filters for improved image quality using the Image Enhancement module.
  - IQA: Checks the image quality and indexes images using the Image Quality Assurance module.
  - ImageExp: Exports images to a back-end system using the Image Export module.
  - IndexExp: Exports index data to a back-end system using the Index Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.

- `Index5.ipp`
  - Scan: Scans or imports document pages into the InputAccel Server using the ScanPlus module.
  - IQA: Checks the image quality and indexes images using the Image Quality Assurance module. Rejected images are routed to RescanPlus. Accepted images are routed to Image Export.
  - Rescan: Rescans rejected images using the RescanPlus module. After the rejected image is rescanned, it is routed to Image Quality Assurance. If an operator does not rescan the image, it automatically proceeds to Image Export.
  - ImageExp: Exports images to a back-end system using the Image Export module.
  - IndexExp: Exports index data to a back-end system using the Index Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Index6.ipp`
  - Scan: Scans or imports document pages into InputAccel Server using the ScanPlus module.
  - IE: Applies image enhancing filters for improved image quality using the Image Enhancement module.
  - IQA: Checks the image quality of every fifth image using the Image Quality Assurance module. Rejected images are routed to RescanPlus. Accepted images are routed to Image Export.
  - Rescan: Rescans rejected images using the RescanPlus module. After the rejected image is rescanned, it is routed to Image Quality Assurance. If an operator does not rescan the image, it automatically proceeds to Image Export.
  - ImageExp: Exports images to a back-end system using the Image Export module.
  - IndexExp: Exports index data to a back-end system using the Index Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.

- `Index7.ipp`
  - Scan: Scans or imports document pages into InputAccel Server using the ScanPlus module.
  - IP: Applies image enhancing filters to images, including the barcode filter to read any barcodes on the image (values from the first two barcodes populate indexing zones defined for the Image Quality Assurance module.
  - QA: Checks the image quality and edit barcode values using the Image Quality Assurance module. Rejected images are routed to RescanPlus. Accepted images are routed to Image Export.
  - Rescan: Rescans rejected images using the RescanPlus module. After the rejected image is rescanned, the image routes to Image Quality Assurance. If an operator does not rescan the image, it progresses to Image Export.
  - ImageExp: Exports images to a back-end system using the Image Export module.
  - IndexExp: Exports index data to a back-end system using the Index Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Index9.ipp`
  - Scan: Scans or imports document pages into InputAccel Server using the ScanPlus module.
  - ImageExp: Exports images to a back-end system using the Image Export module.
  - IndexExp: Exports index data to a back-end system using the Index Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.
- `Index10.ipp`
  - Scan: Scans or imports document pages into InputAccel Server using the ScanPlus module.
  - Index: Indexes images at batch level 0 using the IndexPlus module.
  - ImageExp: Exports images to a back-end system using the Image Export module.
  - IndexExp: Exports index data to a back-end system using the Index Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.

- `PageReg.ipp`
  - Scan: Scans or imports document pages into InputAccel Server using the ScanPlus module.
  - Register: Registers pages to specified templates using the Page Registration module.
  - OCR: Converts images to text using the NuanceOCR module.
  - Rescan: Rescans rejected images. Pages that fail registration (as determined by an error code returned from the Page Registration module) are automatically routed to RescanPlus where an operator can examine the image and either rescan them or send them on to OCR.
  - FileExp: Exports text to a back-end system using theFile System Export module.
  - Done: Reports the batch processing time on the Batch Reconciliation report and indicates that the batch has finished processing using the IADONE virtual module.
  - DeleteBatch: Deletes the batch when it finishes processing using the Multi module.

Chapter 10

Resources

After installing InputAccel, you can access the documentation for each module from **Start > Programs > InputAccel > Documentation > InputAccel Documentation**.

The following guides document general information about InputAccel:

- **Administration Guide**: Documents how to perform other administrative tasks within an InputAccel system.
- **Client Configuration Settings Reference**: Documents system settings that can be used to tune InputAccel performance characteristics.
- **Client Script Engine Guide**: Documents how to create script actions and associate them with specific events defined in each InputAccel module. When the event occurs, the script action is executed.
- **Error Message Reference**: Documents error codes and their meanings.
- **Installation Guide**: Documents installing, configuring, and licensing InputAccel Servers and client modules.
- **Process Developer Guide**: Documents how to plan, create, and test an InputAccel process.

Topics in this section include:

Training

EMC Education supports all EMC business solutions. Certification learning frameworks and training help prepare you to attain technical competency in a product-oriented or functional specialty to optimize your information infrastructure and meet the demands of your business.

Learn more about EMC's service commitment and maintenance options: http://www.emc.com/support-training

Technical Support

EMC's Maintenance and Technical Support helps you access resources whenever you need them to help you get maximum value from your EMC information infrastructure:

- eServices Support Tools: Use EMC's Powerlink web portal for quick online access to solutions, best practices, reference architecture, and more.
- Support Options: Receive rapid response and issue resolution from EMC's global team of experts.
- Secure Remote Support: Ensure high availability with around-the-clock remote monitoring, troubleshooting, and repairs protected by advanced security features.
- Disk Security Services: Reduce costs, enable compliance, and mitigate risk with Disk Security Services.

Learn more about EMC's service commitment and maintenance options: http://www.emc.com/support-training

Index

N

O

P

Q

R

S

T

U

V