

IBM ImagePlus
Workfolder Application Facility for AS/400



Designing a Work Process

Version 4 Release 1

IBM ImagePlus
Workfolder Application Facility for AS/400



Designing a Work Process

Version 4 Release 1

Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page iii.

First Edition (September 1997)

This edition applies to Version 4 Release 1 of the IBM ImagePlus VisualInfo for AS/400 licensed program, Program Number 5733-A18, and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the proper edition for the level of the product.

Order publications through your IBM representative or the IBM branch serving your area. Publications are not stocked at the address given below.

A form for readers' comments appears at the back of this publication. If the form has been removed, address your comments to:

Information Development
IBM Corporation
Department CGMD, Building 062
PO Box 12195
Research Triangle Park, NC 27709-2195 USA

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1993, 1997. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact SWS General Legal Counsel, IBM Corporation, Department TL3 Building 062, PO Box 12195, Research Triangle Park, NC 27709-2195. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

IBM may change this publication, the product described herein, or both. These changes will be incorporated in new editions of the publication.

Trademarks

The following terms are trademarks of IBM Corporation in the United States or other countries:

- Advanced Peer-to-Peer Networking, APPN
- Application System/400, AS/400
- BookManager
- Client Access, Client Access/400
- COBOL/400
- Facsimile Support/400
- IBM
- ImagePlus
- Operating System/2, OS/2
- Operating System/400, OS/400
- VisuallInfo

About This Book

This book shows you how to analyze your enterprise's work processes so you can create your own automated work management process, using either Work Management Builder or AS/400 5250 panels.

Who Should Use This Book

This book can help application programmers who are writing line-of-business applications for IBM ImagePlus Workfolder Application Facility for AS/400 (Workfolder Application Facility), a feature of IBM ImagePlus VisualInfo for AS/400 (VisualInfo for AS/400).

Refer to the *IBM ImagePlus Workfolder Application Facility for AS/400 User's Guide for the Work Management Builder* for details about the skills you need and how to use Work Management Builder.

How This Book Is Organized

This book contains the following information.

- Chapter 1, "Understanding Work Process Management" on page 1 explains work management concepts, including overall workflow and types of routing.
- Chapter 2, "Analyzing Your Current Workflow Process" on page 4 describes how to evaluate and track the current flow of documents, using logs to help do this.
- Chapter 3, "Workbaskets and Profiles" on page 8 explains how to use workbaskets, the various types you can choose, rules that apply, how to associate action lists, how to assign function keys, function key options, and how to link action lists with decision points.
- Chapter 4, "Understanding Collection Points" on page 20 covers the concepts behind collection points and how they relate to the work management process, how and why you would use them, events, event lists, event types, and advanced concepts for handling routing and work package instances.
- Chapter 5, "Understanding Decision Points" on page 30 covers decision points and how they fit into the work management process, how and why you would use them, variables, alternatives, and advanced concepts for handling work package instances.
- Chapter 6, "Example of a Sequential Routing in a Medical Claim Verification Process" on page 39, Chapter 7, "Example of Multiple Routing in a Medical Claim Verification Process" on page 49, and Chapter 8, "Example of Advanced Multiple Routing in a Mortgage Loan Process" on page 68 provide step-by-step examples that show how you can use what you have learned about work management in real-life scenarios by identifying the important workflow elements

Where to Find More Information

The following IBM documents contain information that you might find helpful when using VisuallInfo for AS/400.

For a list of other related publications, see “Bibliography” on page 104. Request copies of IBM publications from your IBM representative or from the IBM branch office serving your area.

IBM ImagePlus VisuallInfo for AS/400

When you order VisuallInfo for AS/400, you receive the following printed publications as part of the VisuallInfo for AS/400 license.

- *IBM ImagePlus VisuallInfo: Client for Windows User's Guide*, SC31-9052
- *IBM ImagePlus VisuallInfo for AS/400: Licensing Information*, GC34-4589
- *IBM ImagePlus VisuallInfo for AS/400: Planning and Installation Guide*, GC34-4585
- *IBM ImagePlus VisuallInfo for AS/400: System Administration Guide*, GC34-4583

The remaining books are shipped in softcopy format only.

- *IBM ImagePlus VisuallInfo: Messages and Codes*, SC31-9065
- *IBM ImagePlus VisuallInfo for AS/400: Application Programming Guide and Reference*, SC34-4586

IBM ImagePlus Workfolder Application Facility for AS/400

When you order the Workfolder Application Facility feature of VisuallInfo for AS/400, you also receive the following printed publications.

- *IBM ImagePlus Workfolder Application Facility for AS/400: Planning and Installation Guide*, GC34-4624
- *IBM ImagePlus Workfolder Application Facility for AS/400: System Administration Guide*, GC34-4625

The remaining books are shipped in softcopy format only.

- *IBM ImagePlus Workfolder Application Facility for AS/400: Designing a Work Process*, SC34-4588
- *IBM ImagePlus Workfolder Application Facility for AS/400: Programming Interfaces Guide*, SC34-4590
- *IBM ImagePlus Workfolder Application Facility for AS/400: User's Guide*, SC34-4584
- *IBM ImagePlus Workfolder Application Facility for AS/400: User's Guide for the Work Management Builder*, SC34-4587

About the Softcopy Document Library

In addition to the printed books you receive as part of the VisuallInfo for AS/400 product, the entire document library is available in softcopy. Shipped on a separate document tape or CD-ROM, the library includes books for the Workfolder Application Facility feature. You can also purchase printed copies of any book in the library.

The library is available in the following formats:

- Portable document format (PDF)
- Hypertext markup language (HTML)

Before you can view or print PDF and HTML documents, follow the installation instructions in either the *IBM ImagePlus VisualInfo for AS/400: Planning and Installation Guide* or *IBM ImagePlus Workfolder Application Facility for AS/400: Planning and Installation Guide*.

PDF Files

Using PDF files, you can easily view documents online, as well as select one or more pages that you want to print.

To view PDF documents, you must have a Web browser and the Adobe Acrobat Reader. If you do not already have this tool, you can obtain it free by following the instructions for download from the Adobe home page:

<http://www.adobe.com/prodindex/acrobat/readstep.html>

A variety of national language readers are also available.

Using the Adobe Acrobat Reader, you can do the following:

- View documents that resemble printed pages.
- Easily navigate through PDF books as you would a hardcopy document.
- Use the text search capability by selecting **Find** from the **Tools** pull-down menu.
- Easily print a single page, a range of pages, or an entire document.
- Customize page size, graphics, fonts, and font sizes.

To view or print books from PDF files using the Adobe Acrobat Reader, follow these steps.

1. Start the Adobe Acrobat Reader.
2. Select **Open** from the **File** pull-down menu.
3. Select the location where you installed the document library.
4. Select the book you want to view.

HTML Files

Using HTML files, you can use a Web browser of your choice to view and print HTML books. These resemble printed books yet provide convenient HTML links to help you navigate through the information.

To view or print books from HTML files, follow these steps.

1. Start your Web browser.
2. Select the **Open** or **Open file** choice (usually from the **File** pull-down menu).
3. Select the location where you installed the document library.
4. Select the book you want to view.

BookManager Files

BookManager files for the VisualInfo for AS/400 document library will be available in the first half of 1998 on CD-ROM as part of the AS/400 softcopy collection kit. To view or print books using BookManager, refer to the books that accompany your softcopy collection kit.

Contents

Notices	iii
Trademarks	iii
About This Book	iv
Who Should Use This Book	iv
How This Book Is Organized	iv
Where to Find More Information	v
IBM ImagePlus VisualInfo for AS/400	v
IBM ImagePlus Workfolder Application Facility for AS/400	v
About the Softcopy Document Library	v
Chapter 1. Understanding Work Process Management	1
Using Information Routing to Define Workflow	1
What is Routed	2
Chapter 2. Analyzing Your Current Workflow Process	4
Keep a Daily Log	4
Analyzing Daily Log Information	5
Collecting Information from Individual Employee Logs	6
Collecting Information from a Group of Employee Logs	6
Chapter 3. Workbaskets and Profiles	8
Choosing a Workbasket	8
Specifying Workbasket Rules and Profiles	9
Defining Action List Options and Function Keys	11
Chapter 4. Understanding Collection Points	20
Collection Points in Work Management	20
What is a Collection Point?	21
What Are Events and Event Lists?	21
Events and Event Types	22
Advanced Concepts	24
Terminology	24
Routes	25
Work Package Instances	26
Additional Information	28
Chapter 5. Understanding Decision Points	30
Decision Points in Work Management	30
What is a Decision Point?	30
Terminology	31
About Variables	31
Alternatives	31
Advanced Concepts	32
Routes	32

Work Package Instances	34
DECIDE Command	35
COPY Command	36
Additional Information	36
Chapter 6. Example of a Sequential Routing in a Medical Claim Verification Process	39
Determining What Workbaskets Are Needed	39
Identifying the Flow of Work Across the Process	43
Defining the Medical Verification Process	43
Workbaskets and Profiles Used in the Medical Claim Verification Process	44
Action List Definitions Used in the Medical Claim Verification Process	46
Work Process Definition for Medical Claim Verification	47
Chapter 7. Example of Multiple Routing in a Medical Claim Verification Process	49
Determining What Workbaskets Are Needed	49
Identifying the Flow of Work Across the Process	55
Defining the Medical Claim Verification Process	56
Workbasket Definitions Used in the Medical Claim Verification Process	56
Action List Definitions Used in the Medical Claim Verification Process	61
Work Process Definition for Medical Claim Verification	64
Chapter 8. Example of Advanced Multiple Routing in a Mortgage Loan Process	68
Determining What Workbaskets Are Needed	68
Determining Contents of a Work Package	68
Determining Where the Documents Originate	68
Determining the Roles Each Employee Performs	69
Determining Security Restrictions and Tracing Activities	71
Determining the Trace Requirements for Each Workbasket	72
Determining the Actions Associated with Each Task Employees Perform	73
Identifying the Flow of Work Across the Process	75
Determining the Final Product of the Work Process in Your Department	75
Determining the Path a Work Package Takes from Start to Finish	75
Determining Where in Your Work Process a Work Package Must Wait	76
Determining Maximum Time for Work Process Step	77
Determining Any Special Handling or Routing Decisions	77
Determining Whether the Action Is Manual or Based on a Program	78
Determining Where Work Package Contents Go When the Work Process Is Complete	78
Defining the Mortgage Loan Process	78
Workbaskets and Their Profiles	78
Action Lists and Their Types	86
Work Process Definition for Mortgage Loan Process	92
Other Objects Needed to Define the Work Flow	100
Optimizing Workbaskets, Action Lists, and Work Processes	100

Appendix A. Case Processing Functions and Work Management Equivalents	102
Bibliography	104
IBM AS/400 Publications	104
Communications and Connectivity	104
Languages	104
Planning, Installation, and Migration	104
Programming	104
System Use	104
System Management	105
IBM Client Access	105
Windows	105
OS/2	105
IBM Data Stream and Object Architectures	105
IBM Facsimile Support for OS/400	105
IBM ImagePlus VisualInfo	105
IBM ImagePlus Workstation Program	105
Windows NT	105
OS/2	105
DOS for Windows	105
IBM ImagePlus Intelligent Forms Facility	105
IBM ImagePlus Intelligent Forms Assist	106
IBM Personal Communications	106
IBM 3995 Compact Optical Library Dataserver	106
Direct-Attached	106
LAN-Attached	106
Both Direct- and LAN-Attached	106
Glossary	107

Chapter 1. Understanding Work Process Management

This chapter introduces you to general work process concepts. You will learn how to outline the steps needed to begin analyzing your enterprise's work process at the employee level and then the departmental level. Because work management provides rich new function, you will also learn about such new concepts and terminology as work processes, routes, workbaskets, work packages, collection points, decision points, and action lists. Toward the end of this chapter are step-by-step examples of work processes, from analysis through a fully developed work process that you can enter directly into IBM ImagePlus Workfolder Application Facility for AS/400 (a feature of IBM ImagePlus VisualInfo for AS/400) or create using Work Management Builder.

Work process management can help you automate and control the flow of work in a department or enterprise. A work process defines how individual users or employees perform their work. Work process management determines how work is:

- Defined
- Assigned
- Prioritized
- Distributed among employees

A *process* is a definable and repeatable sequence of events: It begins with an event, includes activities, and ends with an event. How information flows or is routed through your company determines its *work process*. By identifying a sequence of events, you can easily tell what happens first, what happens next, and so on.

Although work process software cannot fully automate your office, you can use it to automate the flow of information through your office. Using work process software, you can graphically represent the flow of information through your office—and the decisions and actions on that information—rather than representing every action your employees perform. For example, an employee places a document in an envelope, puts a stamp on the envelope, and mails the envelope. In the system you would show only the document being mailed out.

Using Information Routing to Define Workflow

When analyzing the elements in a work process, you need to identify the employees, the information that employees handle, and the path the information follows as it passes from one employee to another.

Work processes can contain several types of routing:

Sequential routing

Work is routed from one employee or location to another employee or location. After the first employee works with an item, the employee sends that item to the next employee. Only one employee performs work at a time, one employee after another.

In some cases, when additional information is required before work can continue, the work item can be placed in *waiting* or *pending* state until the additional information is

received. At that point, all new and waiting information is transferred to the appropriate employee so that work can continue.

Parallel or concurrent routing

Work is routed at the same time to different employees, who might do different things with that same work item. Work management software lets you coordinate the work by waiting for all necessary tasks to complete, bringing the results together, and sending the coordinated work to the next employee or location in the process.

Different employees can receive the work at the same time and all employees who received the work item can perform their work simultaneously without waiting for the other employees to complete their work. Work management software waits for all employees to complete individual work before sending the coordinated work to the next step in the process.

Conditional routing

An employee, a line-of-business application, or information associated with the work itself determines the processing of work. Conditional routing sends the work item along a predetermined path based on the decision.

Exception Processing or Special Handling

Alerts or overrides interrupt the normal processing of work, or exception handling flags the process for special review. Work management software can also predetermine the exception routes. Based on your analysis of your process, you decide and establish in advance the conditions under which a work item is sent along an exception route. You can make options available for the employee to move a work item into a different predefined path from the original path the work item was following.

What is Routed

The information that is routed in Workfolder Application Facility is a *work package*. It contains information your employees process and can be one or more documents. Similar to a folder, it exists only temporarily—for the duration of a process. It represents a unit of work.

A work package can be created when a document is scanned into the system and the document starts on its processing route using Workbasket Types 1 through 5, as described in Table 3 on page 8. You can also create a work package using other methods, such as the following:

- Using work management APIs (BEGPRCS, BEGWPRCS, and CRTWP)
- Using case or file cabinet processing functions
 - Selecting Reassign (F7) assigns an existing case to a work package and begins a process.
 - Selecting Option 21 (Begin process) assigns an existing file cabinet document to a work package and begins a process.

Documents in a work package are stored in a file cabinet or a case before the work package is created. Because the work package is temporary, when all activity on it is

completed, it is removed from the system in one of the ways shown in Table 1 on page 3. In every case, the work package is removed, but you can still access its contents by case or file cabinet processing.

<i>Table 1. Logging Levels for Work Packages for Historical Purposes</i>	
Command	Case Status
STOP	U (unqueued)
CLOSE	C (closed)
QUEUE	A (active); the queue to which the case is assigned is the one specified as the first parameter of the QUEUE command.

So the system removes the work package when a work process ends. The documents and the related information that were maintained with the work package are stored and are not removed.

Chapter 2. Analyzing Your Current Workflow Process

Analyzing your current workflow so you can efficiently use work management software requires some preliminary planning, which is described in this chapter.

Because a work process is a flow of work, to create a process flow you must first define how documents move through your department or group. To define your current business process and how information and documents are worked in your department, go through the activities given below. Even if you are sure that you understand the current business process in your department or group, follow the steps. You might uncover something new about how employees really perform their work.

Keep a Daily Log

The first step is to fully understand the activities of each employee and the role the employee performs in your organization. In the following analysis steps, we talk about individual employees. However, it is really the role that the employee fulfills that is critical to understanding your process.

If you have never attempted to completely map the flow of work in your business process, have your employees keep a daily written log of the following:

- Where the work or documents originate and who gives the employee the work
- The type of documents or information that the employee handles
- The type of documents that accompany the main document that the employee is working with (a complete list of documents the employee is sent)
- Whether the employee has to wait for additional information to work with the documents (and where that information comes from, how long the employee waits, any special conditions)
- The actions that the employee performs on those documents or with that information
- The information that affects the employee's decision to send documents along to one employee or location versus another employee or location
- The department or location the employee sends documents to when the employee is finished with the documents, or what the employee does with the information.

Your experienced employees should perform the logging and initial definition of daily activities, because, even if you do have a formal written process that everyone should be following, exceptions that might have been overlooked in a previous definition occur occasionally.

Collect the logs after a week or a month, depending on how much variability you have in your process.

Because a process is a repeated sequence of activities leading up to the delivery of a product, review your employees' logs for items and activities that repeat. In addition, make special notes of any singular actions or items that do not repeat. Such exceptions are important to identify correctly otherwise the process flow is incomplete and inadequate. If not accounted for in the complete process analysis, the rework due to any exceptions might exceed any benefits that you would normally get from automating your process.

Table 2 shows an example of one format you can use as a daily log.

<i>Table 2. Daily Log for an Employee's Flow of Work</i>	
Category	Details
If the document is in a folder, the name of the folder	
The name or type of the document that you work on	
Other documents in that folder	
The originator: the employee or department that sent the documents to you	
The major action you perform on the document, your role	
The information you need to complete work with the document	
The actions you perform on the document	
Anything you wait for in order to work on the document	
Changes to the document's priority, security, and so on	
Time criteria for processing the document	
Security level of document and folder	
Special handling (rerouting, decisions)	
Tracing or tracking	
Who the document is sent to next (the recipient)	

Analyzing Daily Log Information

For each individual employee or group of identical employees, identify the actions and documents that the employee or group processes, which is the basis for each workbasket. Analyze employee logs both individually (for employees with identical roles) and for groups (across all groups of employees). Group logs are the basis for workflow across your entire department process. Identify the following types of information from the logs:

- Employee role

- Folder contents (work package contents)
- Document-related information
- Employee actions on the documents
- Flow of work, which includes the following:
 - Where information originates
 - Where the employee sends information after finishing the work
 - Special handling (rerouting, waiting periods, and decisions)
- Special computerized information (whether or not it is automated)

Collecting Information from Individual Employee Logs

Follow these steps for individual employee logs.

1. List the documents and their types related to the process that employees work with.
2. List the origination point of those documents.
3. List the role of the employee (what the employee does in the process) such as reviewing or approving a document. Include the general actions related to the role (what the employee does to a document). Identify a clearly defined set of job responsibilities that can be assigned separately to one employee. Even though an individual employee might be doing many tasks, you should separate the individual tasks so that later you can separate the tasks or assign one or more of the tasks to another employee or group of employees.
4. List any security restrictions concerning the documents.
5. List all of the specific actions that occur to each document that an employee handles. Include every action, no matter how minor:
 - Viewing the document
 - Photocopying the document
 - Making notes about the document or its contents
 - Rerouting the document based on special handling rules
6. List any activities that you have to track in a special log, history, or trace file.
7. List whether an employee must wait for other documents or must wait for other information before continuing work with that document.

Collecting Information from a Group of Employee Logs

Follow these steps, using all employee logs.

1. List the main work product (output) of the department. The main work product is the goal to which all activity in your department leads. It can serve as the title of your process.
2. List paths or routes that each document follows (how it gets into your process and how it gets out).
3. List any changes that occur to the priority of a document as it progresses through the process.

4. List any points in the process when one or more employees must or can handle a document at the same time and how those documents later become coordinated. Determine whether only one employee at a time can work on a document or whether several employees can work on it at the same time.
5. Identify whether the action is a manual action or whether the action is based on an automated program that you have written.
6. Identify the time duration of each step. Identify any criteria that you must meet for processing time on each document or action.
7. Identify any special handling that occurs in the process, for example:
 - Sending the document to a special team for handling
 - Returning the document to the employee who sent it out
 - Making decisions about where the document goes next
8. List what happens to the document after the department is finished working on it.

Subsequent chapters contain several examples using individual and group employee logs to illustrate this process analysis.

Chapter 3. Workbaskets and Profiles

This chapter contains basic information about workbaskets and their profiles. After reading this chapter, you can use the various predefined workbaskets.

Think of a *workbasket* as a queue of work you create for one or more employees to use or as the in-basket from which employees take work. Work assigned to a single workbasket can be worked by as many employees as you have assigned to that workbasket. *Work packages* are the individual pieces of work in Workfolder Application Facility. Each represents a unit of work and can contain one or more documents.

Workfolder Application Facility supports six predefined workbaskets and has a seventh type that a system administrator can create. See Table 3.

Workbasket Type	Name/Purpose	Description
1	Scan Documents	Accepts documents to be scanned into the system and starts them on a work process. Consists of the documents waiting to be scanned.
2	Index into Folders	Accepts documents to be entered into the system and indexed into folders or cases.
3	Index into File Cabinets	Accepts documents to be entered into the system and indexed into file cabinets.
4	Scan and Index into Folders	Accepts documents to be scanned into the system, indexes them into folders, and optionally starts them on a work process. Consists of the documents waiting to be scanned.
5	Scan and Index into File Cabinets	Accepts documents to be scanned into the system, indexes them into file cabinets, and optionally starts them on a work process. Consists of the documents waiting to be scanned.
6	Work with Work Packages	Displays a list of information assigned to a work package, such as documents in the folder or information related to a single document in a file cabinet.
50-99	Customer-defined	The system administrator defines the user interface and function of this workbasket type.

Choosing a Workbasket

Table 4 on page 9 describes the types of workbaskets you can choose, based on the type of Workfolder Application Facility system you have.

To support	Choose Workbasket
Scanning and indexing	Type 1 through 5: <ul style="list-style-type: none"> • If a separate group of employees only scan documents into the system, use Type 1 and then select either a Type 2 or 3 workbasket, depending on whether you have folders or file cabinets on your Workfolder Application Facility system. • If a group of employees both scan and index, select either a Type 4 or 5 workbasket, depending on whether you have folder or file cabinets on your Workfolder Application Facility system.
General work on work packages, where users look at a work package's contents, make notes about the documents, and so on	General Type 6 work with work packages workbaskets
Special needs	You can create up to 50 other types of workbaskets. Refer to the <i>IBM ImagePlus Workfolder Application Facility for AS/400: System Administration Guide</i> for more information about creating your own customer-defined (Type 50-99) workbasket.

Specifying Workbasket Rules and Profiles

Workbaskets have associated rules. You specify the rules for each workbasket at the same time you create the workbasket. Each different type of workbasket has a different set of rules that apply. Depending on the type of workbasket you create, you must specify the following:

Security level of the workbasket

The security level of a workbasket determines whether a user can access that specific workbasket. For example, if you assign a security level of 50 to the user in the user profile and the workbasket has a security level of 75, the user cannot work any item in the workbasket.

Whether work can be reassigned to the workbasket

When an employee's day-to-day decisions are the basis for special handling or exception processing, you might want to let the employee reassign a work item to another workbasket for special handling. You can do this by specifying that the receiving workbasket can have work reassigned to it. However, for normal, everyday routing decisions along predictable paths, use the continue function to move information to the next workbasket.

You can also prevent work from being reassigned to a specific workbasket. For example, it might not make sense to let employees reassign work to a workbasket where indexing is performed.

Whether user activity should be logged and what type of logging should be performed

When you specify that logging should be performed at a workbasket, you can specify whether every access to the workbasket is logged and whether individual actions the user performs are logged. This information is placed in the document log file. To retrieve logged information, use the key fields defined below and the AS/400 query function.

If logging is performed, you can specify when it actually occurs. See Table 5.

Logging Level	Stored in System
1	After the user leaves the workbasket
2	After the user performs the function but before the user leaves the workbasket

Whether you want to log the history of the work package

When you specify that you want to log the history of the work package, you can specify three different levels of logging. See Table 6.

Logging Level	Stores Number of Times for Each Activity
1	<ul style="list-style-type: none">• A work package is assigned to a workbasket• Retrieved from a workbasket• Removed from a workbasket
2	Stores Level 1 information and also the function keys used
3	Stores Level 2 information and also the option codes used at the workbasket

Up to three report keywords for finding logged information when querying log file

The keywords are stored with the other information you specified to be logged.

How work packages are listed or presented to users

First in, first out (FIFO), last in, first out (LIFO), or priority.

Name of the default action list

Specify the name of the action list to be used with this workbasket. This is the default action list. Because it is the default, you do not specify the action list again on the routing panels.

If you wanted to perform special processing on selected work packages in a work process, you can create two different action lists: one as a default, and one for special processing. For example, you can send a work package to a decision point that evaluates information about the work package. There might be a value associated with the claim that flags it as being a claim for more than \$100,000. At your decision point, you send the work package to one route or another, depending on the value of the claim. Each of those routes then uses the WORK VERIFY command, but the route for under \$100,000 might enable the Issue Check function

using the default action list. The route for over \$100,000 might use an OVER100 action list that contains a special action to be taken for checks over \$100,000. You then use the WORK VERIFY OVER100 command to use the special action list. The action list OVER100 overrides the default action list.

Defining Action List Options and Function Keys

When you create a work process, you assign a list of actions to each workbasket. The action list must be a comprehensive list of all actions performed on a work package or on an individual document in that work package. If the action is not listed on the action list for a workbasket, the employee cannot perform that action.

So, in addition to the unique major tasks performed at each workbasket on the work package and documents, you must include a list of all other activities you want your employees to perform on the documents. See “Workbasket Actions” for a list of all activities that can be assigned in an action list. For example, keep in mind that your employees perform some basic actions, so you should consider supplying actions that let them do the following:

Exit

Leaves the work package without changing anything

Continue

Leaves the work package after the employee has completed work on the work package, letting the work package continue through the work process

View work package

Sends the contents of the work package to the image display so the employee can view all of the contents

View documents

Sends a selected individual document in the work package to the image display so the employee can view it

Comments

Creates, changes, or deletes a comment for a work package

Workbasket Actions

Each workbasket that you define can have identical actions, or each workbasket can have a different set of actions. You specify which actions you want your employees to perform on the work package or its contents by creating action lists.

Use function keys and option codes to specify exactly which actions a Workfolder Application Facility user should be allowed to perform. If you do not specify a specific action in an action list, the user of the workbasket associated with that action list cannot perform that action. For example, if you do not specify a continue function for an action list, Workfolder Application Facility users cannot send a work package along the specified routing. The work package remains in the workbasket until the user reassigns it using the reassign function. As another example, if you do not specify a display document option code, users cannot select individual documents from the work package to be displayed on the image monitor. Keep the following in mind:

- If you have only a single document, it is automatically displayed on the monitor.
- If you have multiple documents, your users must select one to be displayed.

When you create an action list, the actions you assign are displayed on Workfolder Application Facility panels so users can have that function or option available as an action.

Functions correspond to function keys. They work on the entire contents of work packages. Options correspond to option codes that are shown on the top of list panels. Option codes work on individual documents within a work package, not the entire contents of a work package.

The actions available for Index type and Work type action lists are different. The definition of the options and functions are listed in “Definition of Action List Options” on page 13 and “Definition of Action List Function Keys” on page 14. Following are the actions available in each type of action list:

- Functions for index type action lists
- Options for work type action lists
- Functions for work type action lists

No options are available in index type action lists because each document is already acted upon separately. Note that the Type 1 workbasket (Scan documents) does not have an action list because Workfolder Application Facility performs all actions on the document.

Following are the function keys available for Index Type action lists used in Type 2 through 5 workbaskets:

- Cancel
- Continue
- Delete
- Exit
- Prompt
- Reassign
- Retrieve
- Skip
- Customer-defined

Following are the options available for Work Type action lists used in Type 6 Workbaskets:

- Change document
- Copy document
- Delete document
- Display document
- Document details
- Export document
- Fax document
- Modify document
- Move document

- Print document
- Reindex to queue
- Reindex to workbasket
- Retrieve document
- Customer-defined option
- View locations
- Work versions

Following are the function keys available for Work Type action lists used in Type 6 Workbaskets:

- Add form
- Begin process
- Cancel
- Close
- Comments
- Continue
- Display folder
- Exit
- Export folder
- Fax folder
- Force continue
- Force end
- Import to folder
- Pend
- Print folder
- Reassign
- Refresh
- Release key
- Reverse order
- Search work
- Skip
- Customer-defined
- View history

Definition of Action List Options

Following are the definitions of the option codes, in alphabetical order, that you can use in action lists to let end users perform the specified actions. A Workfolder Application Facility panel is displayed to let users complete their transactions, except for the customer-defined options that the system administrator must write.

Option name Definition

Change document

Changes selected information about the document, such as:

- Current file cabinet keywords and document type for file cabinet documents.
- Current document type for case documents.

Copy document

Copies the selected document into a user-specified case, file cabinet, or both.

Delete document

Deletes the selected document from the current work package.

Display document

Displays the selected document on the image monitor.

Document details

Displays information about the selected document, such as file cabinet keywords and document creation date. This option is available only for file cabinet documents. If the system administrator lists it on a case document, users receive a message informing them that the function is unavailable.

Edit document

Displays the selected document on the image monitor for editing.

Export document

Creates an export request for the selected document.

Fax document

Creates a fax request for the selected document.

Move document

Moves the selected document into either a user-specified case or file cabinet.

Print document

Creates a print request for the selected document.

Reindex to queue

Moves the selected document from the current work package into the reindex queue

Reindex to workbasket

Moves the selected document from the current work package and assigns it to a customer-specified, indexing workbasket.

Retrieve document

Creates a request to retrieve the selected document from secondary storage.

Customer-defined option

Sends information about the selected document and the current work package to a user exit, from which custom code can be executed.

View locations

Displays a list of all cases and file cabinets in which the selected document exists.

Work with versions

Lists all versions of the selected document. Users can work with document versions individually from this secondary panel. The primary panel always displays only the most current version of a document.

Definition of Action List Function Keys

Here are the function keys, in alphabetical order, that you can use in action lists to let users perform the specified actions. A Workfolder Application Facility panel displays to let users complete their transactions, except for the customer-defined options that the system administrator must write.

Function name Definition**Add form to folder**

Selects a form from a list of forms to be added to the current case.

Begin process

Creates a new work package—which contains the same information as the current work package—and assigns the new work package to a user-specified process.

Cancel

Returns the current work package to the current workbasket without sending the current work package on in the process.

Close

Closes the folder work package and removes it from the current process.

Comments

Creates, changes, and deletes comments for cases.

Continue

Removes the current work package from the current workbasket and releases the work package to continue on to the next predefined step of the current process.

Delete

Deletes the selected document from the current process.

Display folder

Displays all of the documents in the current folder work package on the image monitor.

Exit

Returns the current work package to the current workbasket. The work package remains in the current workbasket.

Export folder

Creates an export request for each document in the current folder work package.

Fax folder

Creates a fax request for each document in the current folder work package.

Force continue

Removes the current work package from the current workbasket and continues the work package to the next predefined step of the process. If the current work package is located at a collection point, the work package starts on the exception route defined for that collection point. The collection point is marked as being satisfied and no longer waits for any events to occur.

Force end

Removes the current work package from the current process, regardless of the location of the work package within the process.

Import to folder

Displays panels that lets documents be imported to the current folder work package.

Print folder

Creates a print request for each document in the folder work package.

Pend

Pends the current work package according to customer-specified reasons.

Prompt

Displays a list of values on a secondary panel for the input field on the primary panel. Users can select one of the values to fill in the input field on the primary panel. For example, if the input field is **Document type**, the prompt key displays a panel that lists document types that can be selected to fill in the input field.

Reassign

Removes the current work package from the current workbasket and assigns the work package to a user-specified workbasket.

Refresh

Clears the current panel of any user input.

Retrieve

Fills in the values in the input fields of the current panel with values the user entered on the previous display of the same panel.

Release

Releases a pending work package according to user-specified reasons.

Reverse order

Reverses the displayed order of the documents in a folder work package.

Search work

Accesses either case or file cabinet functions without having to exit from working with the current work package.

Skip

Returns the current work package to the current workbasket and moves on to the next work package without sending the first work package on in the process. This function key is useful when a user is working with work packages in the order that the system displays the work to the user. It lets the user bypass the current work package and move on to the next work package.

View history

Displays the historical data that pertains to the current work package.

Customer-defined

Sends information about the current work package to a user exit, from which custom code can be executed.

Using the Action List Functions to Make Routing Decisions

The action list has several methods to support routing decisions:

- Continue function
- Reassign function
- Begin process

Routing Along Predefined Paths Using the Continue Function: One of the most important features of action lists that you will want to exploit is the feature supported by the continue function. This function lets a user press a function key that indicates the work is done, releasing the work package so it can continue along the predefined path.

Employees can use the continue function extensively in workbaskets where they have to make decisions about routing. However, you no longer need to supply users with lists of people or routes to send the information on to. You can set up an action list with multiple continues, each with its own prompt label, which will send the work package to the correct group of users, using the continue function in conjunction with a decision point.

See Table 7 to learn more about using the continue function. The table shows that you can assign the continue function as many times to a single action list as you have function keys available. That means you can actually assign it 20 times—because you must always define an Exit function, F1 is always help, F23 is more options, and F24 is more keys. In this example, the continue function has been selected three different ways, defined the workbasket that the work package will be sent to, assigned a prompt name that will actually appear on the 5250 panels, and assigned the continue function to a key number.

For example, when users press the F6 key on their 5250 panel, the work package they are working on is routed directly to a decision point. The decision point performs the routing based on the function key number that was assigned. In this example, F6 sends the work package on to the Reject workbasket, where any number of other users can be working.

<i>Table 7. The Continue Function Used in Action Lists</i>			
Function	Route to	Prompt name	Function key
Continue	Send on to reject processing	Reject	F6
Continue	<ul style="list-style-type: none"> • Send on to approval • Send on to quality control 	Approve	F7
Continue	Send on to a review supervisor	Review	F8

Because Workfolder Application Facility supports parallel processing, you can also have the continue function send the work package to two or more different workbaskets at the same time. The F7 (Approve) key was assigned to the continue function. Even though the user of this action list might only press one function key (F7), that action sends the work package to the next step in the process. The next step could be two START commands, which will send the work package to both the Approval team and the Quality Control team.

Based on the actual name you put into the prompt field, you can choose to allow whether your employees are aware of where they are sending the work package. You can choose to set up hidden routing, as in the Quality and Approval team example, depending on how you actually name your prompts. However, the most important

advantage of this type of routing is not the ability to hide the routes, but that users no longer have to know the user ID or name of the person to whom the work package is being routed, meaning they no longer need a list of user IDs for normal processing of information. The process you create defines the routing destinations. See “Routing Along Undefined Paths” for details about special situations where routing occurs along paths that are not defined.

Routing Along Undefined Paths: Even in the best defined process, a work package might need to be sent along a path that has not been previously defined in your current work process. You can use action lists to do this in one of two ways:

- Reassign function
- Begin process function

Undefined Routing Using the Reassign Function: Use the reassign function to let users send a work package to another workbasket that is not defined as the next step in your work process. You can use this function to correct human mistakes in a process. When reassigned, a work package is not removed from its current process.

For example, consider a process that is defined to send work packages from workbasket A to B to C. A user who selects a work package from workbasket B notices that the work needs management approval. The user reassigns the work package from workbasket B to workbasket MANAGER. When the manager completes that work and uses the continue function, the work package continues to the next step of the process that follows workbasket B—in this case, workbasket C. If the work had to return to workbasket B, the manager could have reassigned the work package to workbasket B.

A work package can be reassigned as many times as necessary to workbaskets inside and outside of its current process. The work package always remains associated with its current process. Following a reassign function, the first time a user selects a continue function, work continues to the next step in the process that would normally follow the step from which the work package was first reassigned.

If any of the initial review teams had made a mistake and sent along a document to the next step in the process in the examples in this book), users in the next workbasket could use a reassign function to send the work package back to be reviewed again. (See Chapter 6, “Example of a Sequential Routing in a Medical Claim Verification Process” on page 39, Chapter 7, “Example of Multiple Routing in a Medical Claim Verification Process” on page 49, and Chapter 8, “Example of Advanced Multiple Routing in a Mortgage Loan Process” on page 68.) Even though the examples defined only the correct error-free paths, you could certainly have also set up a predefined path for that action.

Undefined Routing Using the Begin Process Function: The begin process function is similar to a copy function. When a user selects a begin process function, a new work package is created and assigned to the process that the user specifies. The contents of the work package are the same as for the work package for which the user selected the begin process function. The end result of using this function is two work packages with the same contents.

The original work package is not removed from the original process. This also means that work process for one copy of the work package can end independently of the work process for the other copy of this work package.

For example, a single letter is sent from a customer to a charge card department, requesting a correction to a bill and an address change. A user scans, stores, and assigns the letter to the BILLS process. At any point in the BILLS process, a user can select the begin process function to create a copy of the letter and assign it to the ADDRESS process. Now two users can work on the same letter at the same time as the letter moves through two separate processes. This is unlike parallel routing, where you might want the information to be coordinated later. These work processes are independent of each other and will end at different times.

Chapter 4. Understanding Collection Points

This chapter explains the concept and uses of collection points. After reading it, you will have a better understanding of collection points and how they relate to work management.

You use a *collection point* to collect information from different sources before passing the work package on in the process. A collection point holds and manages work packages that cannot be processed completely until additional information has been received.

A collection point performs no action. It merely represents the fact that a work package is waiting for one or more events to occur. The work package could be waiting for a single event, two, three, or even more events of different types.

A collection point waits for events to occur. (For more information on the types of events that can be specified in a collection point, see “What Are Events and Event Lists?” on page 21.) The three most common types of events are based on the following:

Documents	The collection point waits for specific document types.
Routes	The collection point waits for processing to occur in a previous step (workbasket).
Days	The collection point waits for a specific number of days.

In some work processes, several documents of different types might all have to arrive before any of them can be worked. The documents that the collection point is waiting for might comprise an event list. For example, in a loan application, you might need to have a credit appraisal along with the loan application before any formal work can be done on the loan application. The method to represent the fact that several documents must be together before either of them can be worked on is called the collection point.

After the event you have specified has occurred, the work package is then sent along the processing path that you have set up. If the specific event has not occurred in a specified amount of time, the work package is processed along another path that you also set up. This alternative path is referred to as the exception path. The criteria for sending the work package along the exception path is based on the number of days that you specified that the work package should wait before the work package continues through the process.

Collection Points in Work Management

Before you can understand what a collection point is, you at least need to understand the processes in which collection points exist.

In work management, a *process* is a set of routes, workbaskets, collection points, and decision points that are organized in a specific way and through which work packages

move from start to finish. Figure 1 on page 21 shows an example of a simple process diagram.



Figure 1. A Very Simple Process with a Collection Point

The example is simple to keep the meaning of the diagram fairly clear. A work package which enters this process first goes to the collection point. When it leaves the collection point, it continues to the Type 6 workbasket. Finally, when the work package leaves the workbasket, it has completed the process.

What is a Collection Point?

A collection point is a part of a process that work packages can be sent to. But, what is a collection point, why would you want to send work packages to it? And when does a work package leave a collection point once it is there? A collection point is a location in a process where work packages wait until one or more events have occurred.

By sending a work package to a collection point, you stop it from continuing through the process until some set of necessary events have occurred. When all events in a set have occurred, the work package automatically leaves the collection point and continue on to the next step in the process.

What Are Events and Event Lists?

A set of events in a collection point is called an event list. An event list contains one or more events, all of which must be satisfied for a work package to be released from a collection point. The event list also contains the name of the *route*—a sequence of steps in a process—to which a work package is sent when all events in the event list are satisfied. Routes are explained later (see “Advanced Concepts” on page 24).

A collection point can have from 0 to 999 event lists. It is possible to have no event lists in a collection point, because there is a special exception condition which always let a work package exit a collection point. The exception condition is the number of days a work package waits at the collection point. If a work package has not already left a collection point when that number of days has passed, then the work package leaves the collection point and go to the route specified as the exception route for that collection point.

A work package is released from a collection point as soon as all of the events in any one event list have occurred. Event lists within a collection point are not related to each other.

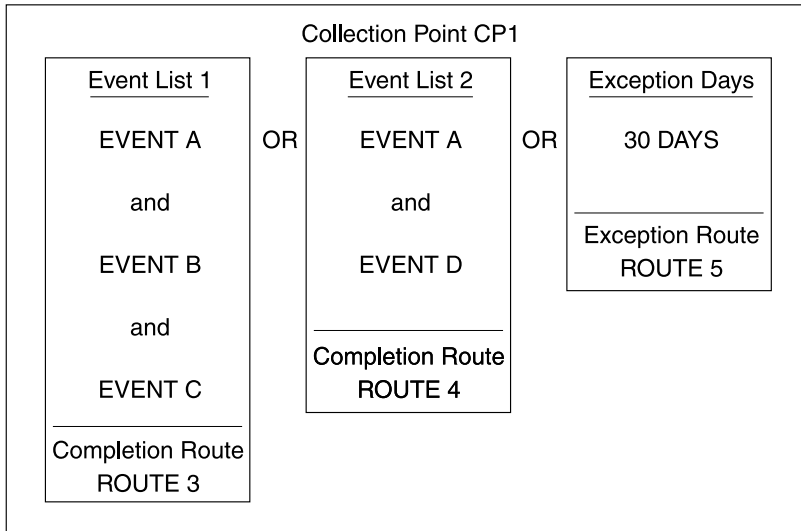


Figure 2. A Collection Point

Suppose a work package is waiting at the collection point shown in Figure 2. If events A, B, and C are all satisfied for the work package, then the work package leaves the collection point and go to Route 3. If events A and D are satisfied, then the work package continues to Route 4. If A, B, and C are not satisfied, and if A and D are not satisfied, then 30 days after its arrival at the collection point, the work package continues to the exception route, Route 5. Note that event A occurs in two event lists. This is perfectly within the rules of collection points.

Events and Event Types

Events have been mentioned throughout this introduction to collection points, so it might be a good idea to explain just what they are. An event consists of an event type and a match criterion. Event types are actually two-digit numbers, but they are usually referred to by words which describe what events of that type are supposed to wait for. Following are the valid event types.

Document event type (01)

An event of type DOC indicates that the collection point is waiting for a specific type of document to be added to a particular work package. If a match criterion of *ANY is defined for a collection point, it matches any document type. When attempting to match an event, Workfolder Application Facility reads through the collection point events in sequence. If you have an event list waiting for a document type DOC1 and a document type of *ANY, it is possible that when DOC1 comes to the collection point, event *ANY is satisfied but not DOC1—depending on the order the event list is defined in the collection point profile.

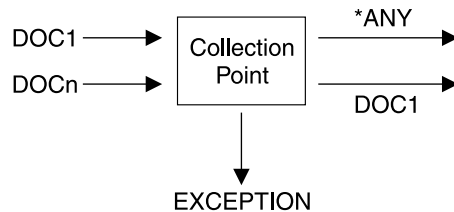


Figure 3. A Very Simple *ANY Collection Point

Which route is taken for DOC1-*ANY, DOC1, or both? The route that is defined first. So what is defined first? When Workfolder Application Facility matches events, it simply attempts to match the incoming event to the unresolved events in the order they were defined. So, if DOC1 and *ANY are defined as collection point events and DOC1 arrives at the collection point, either the DOC1 or *ANY event is matched depending on the order in which they appear. Note that one incoming event can never match with two events in the same event list; there is never more than one route started when a collection point is released.

Route (02)

An event of type ROUTE indicates that the collection point is waiting for an instance of a work package to enter the collection point from a certain workbasket. There is no ROUTE in Work Management Builder. Instead a workbasket is shown as a choice in the event list, which represents the route.

Days (03)

An event of type DAYS indicates that the collection point is waiting until a work package has been in the collection point for a certain number of days. For the days event type (03), the code checks to see if it can satisfy the expired time check. If so, the work package instance is released and the exception route started.

Force release (04)

An event of type FORCE RELEASE indicates that the collection point should be released immediately regardless of the events that have been satisfied and the exception route started. The collection point profile is then examined to determine the collection point exception route to begin and the exception route started.

Customer-defined (50-99)

An event of a customer-defined event type can be whatever the system administrator wants it to be. Processing for the customer-defined event types (50-99) is very similar to that of the document event type (01). The logic flow is identical. The customer-defined event types use the MATCHCP API.

If you had a work package waiting for an event of type DOC, the work package waits at the collection point until a document is added to it. Indicating specific information about an event, which then determines which document, the match criterion field contains a value that compares and matches a waiting event to an event that has occurred. Content of the match criterion depends on the event type of the event, as follows.

DOC

The match criterion is the document type of the document for which the work package is waiting. You can also define *ANY to have any incoming document type satisfy the match criterion.

ROUTE

The match criterion is the name of the route from which the work package instance is arriving at the collection point.

DAYS

The match criterion is the number of days the work package waits in the collection point.

FORCE RELEASE

The match criterion is ignored; the collection point is released regardless of the match criterion.

USER-DEFINED

The match criterion is customer-defined comparison data.

As an example, if a work package is waiting at a collection point, and it is waiting for an event of type DOC with match criterion CLAIM, then the event is only satisfied when a document of DOC type CLAIM is added to the work package.

Advanced Concepts

Before going into greater detail in explaining the workings of collection points, there are a few more work management concepts that need to be understood, as they are crucial to comprehending all the nuances of collection points.

Terminology

You should be aware of a few terms which are heavily used when discussing collection points, and which can be quite confusing if you don't clearly understand them. Notice how they relate to each other.

Matching an event

Finding an outstanding event at a collection point that has the same work package, event type, and match criterion as the event being used for comparison.

Satisfying an event

Removing a matched event from the outstanding events at a collection point. This is always done when an event is matched.

Releasing a work package from a collection point

Removing a work package from a collection point when all of the events in an event list have been satisfied. The work package goes to the route specified as the completion route for that event list.

Activating a work package at a collection point

Creating the outstanding events for a work package at a collection point when an attempt to match an event has failed.

Routes

Take another look at the process diagram you saw earlier. See Figure 1 on page 21.

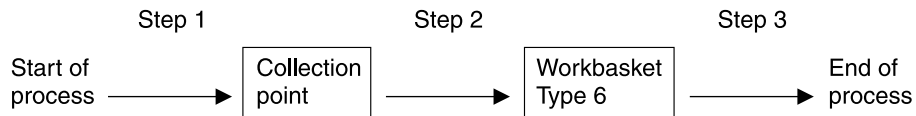


Figure 4. A Very Simple Process with a Collection Point Showing Steps

A work package that enters the process goes to the collection point. That is the first step in the process. In the second step of the process, the work package goes to the Type 6 workbasket. The third and final step in the process is for the work package to leave the process. As you can see, each arrow in the diagram represents one step. In work management, steps are always recorded sequentially in groups. A group of steps is called a route.

A route consists of one or more steps. When a work package enters a process, it is actually entering the first route of that process. As the work package proceeds through the route, each step in the route is performed using that work package, until the end of the route is reached.

One way that a route can end is with a step which sends the work package to a collection point. This type of step, called a COLLECT command, must always be the last step in a route, since a work package leaving a collection point always goes to the new route specified in the satisfied event list of the collection point.

Now you know enough to understand how the process diagram represents the true structure of a process. The example shown in Figure 5 on page 26 shows the process diagram again, this time followed by the actual process objects (routes, a collection point, and a workbasket, in this case) which make up the process.

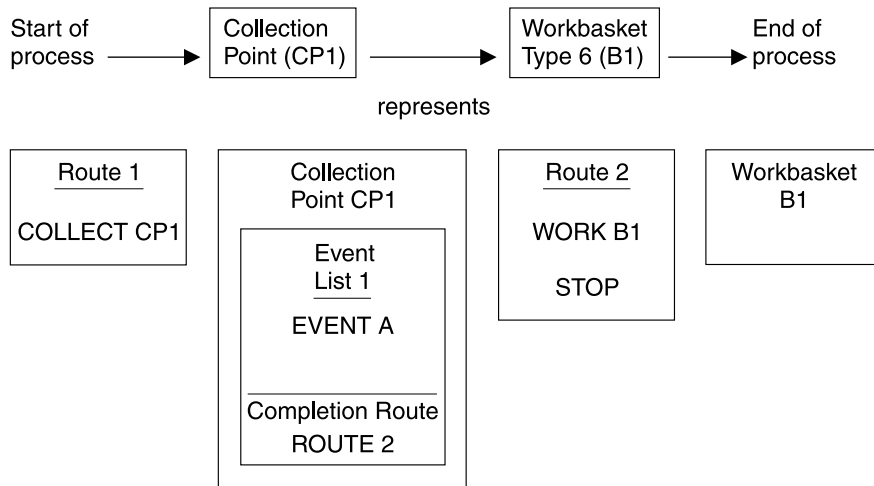


Figure 5. Simple Process with Process Objects for a Collection Point

A work package which enters the process goes to the first step of route 1, which is a command COLLECT to send the work package to the collection point called CP1. When event A occurs, the work package leaves the collection point and continues to route 2. The first step of route 2 sends the work package to workbasket B1. When the work package exits the workbasket, then the next step in route 2, the STOP command, is executed. STOP stops the movement of the work package through the process, ending the process.

Work Package Instances

The simple process diagram that has been used until now involves a work package moving through the route from start to finish in a linear fashion. But what about the process shown in Figure 6?

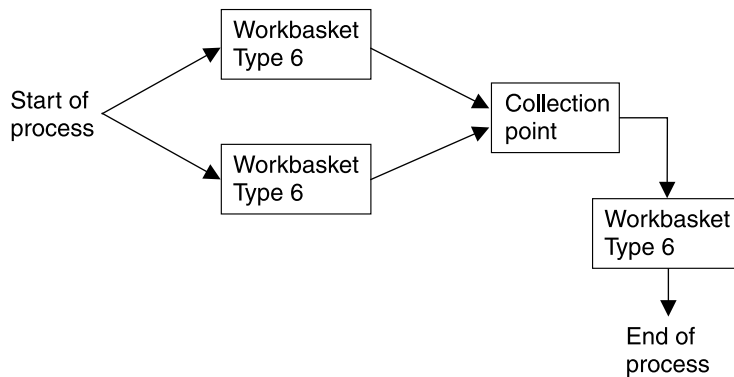
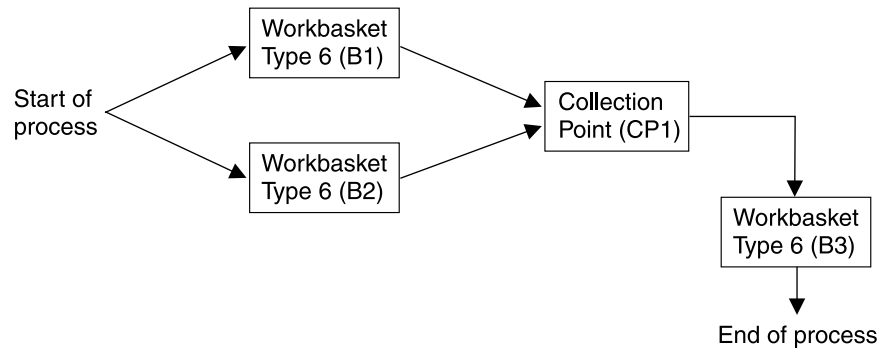


Figure 6. A Parallel Process with Collection Point Coordination

A work package entering this process immediately splits, continuing to two workbaskets at the same time. Actually, a pointer to the work package moves through the system. Such pointers contain the names of the work packages they represent. If information about a work package is needed, the system uses the pointer to determine which work package information to examine. These pointers are called instances of a work package. A work package can have up to 99,999 instances, which means that a work package can be in up to 99,999 different places at once during a process.

One of the event types of collection point events is the type ROUTE, which indicates that a collection point is waiting for an instance of a work package to enter the collection point from that route. The concept becomes clear when the above process is examined. The work package splits into two instances, but is reassembled in the collection point so that only one instance leaves the collection point. To do this, the collection must be able to wait for individual instances of a work package. When all instances arrive at the collection point, the work package continues to the next step in the process.

See Figure 7 for a process diagram.



represents

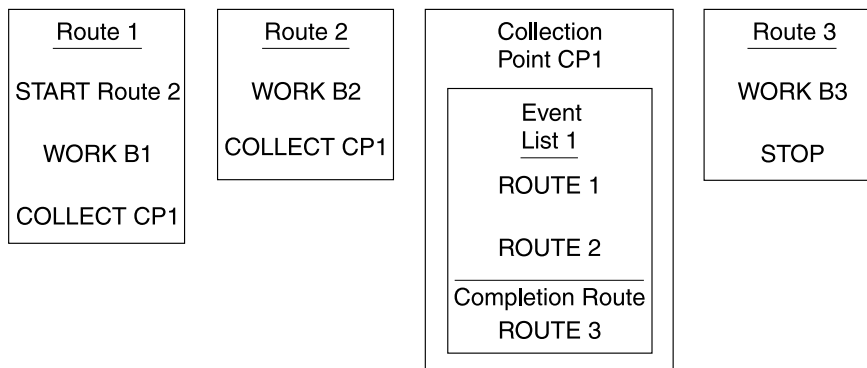


Figure 7. A Parallel Process and Objects with Collection Point Coordination

As illustrated, the second instance of a work package is created by route 1 when the START Route 2 step is performed. Using START is the only way to create additional instances of a work package. Because the new instance must start on a different route, a collection point must be able to identify different instances by looking at the route from which each originates.

The collection point waits for a work package instances to arrive from routes 1 and 2. When both events are satisfied, the collection point releases the work package (actually, a single instance of the work package) to route 3, which ends the process after sending the work package to workbasket B3 for some final work.

Additional Information

The following information can help you better understand how collection points work.

- It is possible to start a work package on parallel routes as a result of releasing a work package from a collection point. When all the events in an event list are satisfied, the completion route is started even if the work package is at a workbasket as part of an active route. The work package is then active at two routes in the process. To avoid this type of situation, be sure you include all routes in the event list so that the collection point is not released until all routes reach the collection point from a COLLECT command.
- The collection point is activated if the collection point code is called to match an event that is not part of the event list—more specifically, if a COLLECT command occurred and a document type is defined with an activate collection point of Y, or if the MATCHCP API is called with an activate collection point of Y. No events are satisfied, and the collection point cannot be released as a result of this event.
- When a collection point is released to the next route or exception route, the values of the variables associated with the route are equivalent to those of the event that activated the collection point. For example, if a route event activates the collection point, the route started by the released collection point has the same variable values as the activating route. If a document event activates the collection point, the route started by the released collection point has variable *BUTTON with a value of 0000000000 and variable *PRIORITY with a value of spaces.
- Search for work packages lets you find work packages based on indexing information or current location in a process. While this function does not find work packages at collection points that a document activated, it does find work packages at collection points that the route COLLECT command activates.
- A document type 1 (create new case) that has an activate collection point flag of N can never be matched at a collection point unless you use the MATCHCP API. For more information about document types, refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: System Administration Guide*.
- Many things can happen as a result of calling the MATCHCP API and the API can only return one return code. You should review the error log file EKD0080 for other possible errors. The MATCHCP API returns the status of what occurred at the collection point: that is an event was satisfied, the collection point was activated, and so on. If the collection point was released and an error occurred during

processing of the completion route, the API does not return an error but logs it to EKD0080, returning a code to notify the caller that the event was matched.

- If the same item in an event list is defined twice—for example, two entries of event type 01, where the document type is DOC1—the specified item must be satisfied twice. One event cannot satisfy more than one event in a single event list.
- If you have a document in a collection point event list, that document can only be satisfied by indexing through workbasket types 02 or 04, unless you use the MATCHCP API. Documents indexed using workbasket types 03 or 05 cannot satisfy a collection point. Documents indexed using any nonwork management indexing function can never satisfy a collection point event.

Chapter 5. Understanding Decision Points

This chapter explains the concept and uses of decision points. After reading it, you will have a better understanding decision points and how they relate to work management.

You use a *decision point* to determine the next step in the work process. Decision points represent choices based on line-of-business information, work management information, and the function key you use.

Decision Points in Work Management

Before being able to comprehend what a decision point is, you first need at least a little understanding of the processes in which decision points exist.

In work management, a *process* is a set of routes, workbaskets, collection points, and decision points which is organized in a particular way, and through which work packages move from beginning to end. The easiest way to describe a process is by drawing a process as a diagram. Figure 8 shows an example of a simple process diagram with a decision point.

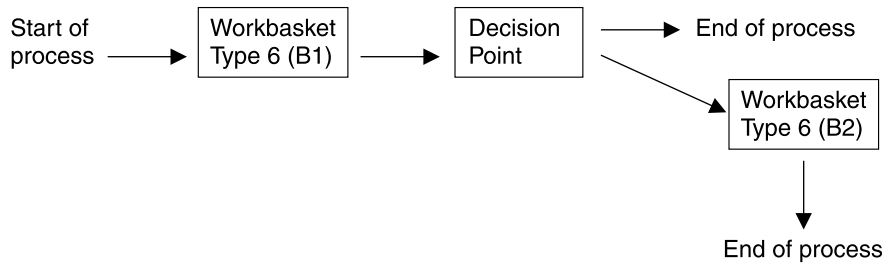


Figure 8. A Very Simple Process with a Decision Point

The example is simple to keep the meaning of the diagram fairly clear. A work package that enters this process first goes to the Type 6 workbasket. When the work package leaves the workbasket, it is sent to the decision point. At the decision point, the work package can be sent to another Type 6 workbasket or the process ends.

What is a Decision Point?

A decision point is a location in a process where work packages are sent to different routes depending on criteria specified by the user. By sending a work package to a decision point, you are asking the system to start a route after analyzing the values of variables associated with that work package. Unlike collection points, there is no waiting that takes place at a decision point.

Terminology

Before going any further, you should be aware of a few terms used when discussing decision points.

- Alternatives are the variable, value, and route combination that determines where the work package goes when it leaves the decision point and what condition caused it to take a particular route.
- Variable is a 1- to 10-character alphanumeric name. With the exception of two system defined variables (*BUTTON and *PRIORITY), variable names are customer-defined.
- Value is a 1- to 10-character alphanumeric string associated with a variable name. For example, variable *PRIORITY could have a value of 971231.

About Variables

A decision point alternative match occurs when a work package entering a decision point meets the conditions (the variable and its value) you define for an alternative. The variable can be one of three types:

***BUTTON**

Lets you determine what happens when a user presses a given function key (or button) at a workbasket before sending the work package to the decision point (using the DECIDE command). The value for *BUTTON can be a number from 2 through 22, for the valid function keys.

For example, if you specify a variable of *BUTTON with a value of 2, when the user presses F2 at the workbasket and sends the work package to the decision point, the conditions for this alternative are met and the work package is sent to the route you specified in the alternative. However, in order to have the work package go on to the next step in the process, the underlying function code must be Continue.

***PRIORITY**

Lets you assign a higher or lower priority to the alternative you are working on. For example, 2 has a higher priority than 1.

Customer-defined

Lets you define your own variables and values for work packages. The customer-defined variable name is 1 to 10 characters, and the variable name cannot start with an asterisk (*). You can use the COPY command or the CHGSYM and QRYSYM APIs to work with customer-defined variables.

For more information about the variables described here, refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: System Administration Guide*.

Alternatives

The diagram in Figure 9 on page 32 illustrates what a decision point looks like.

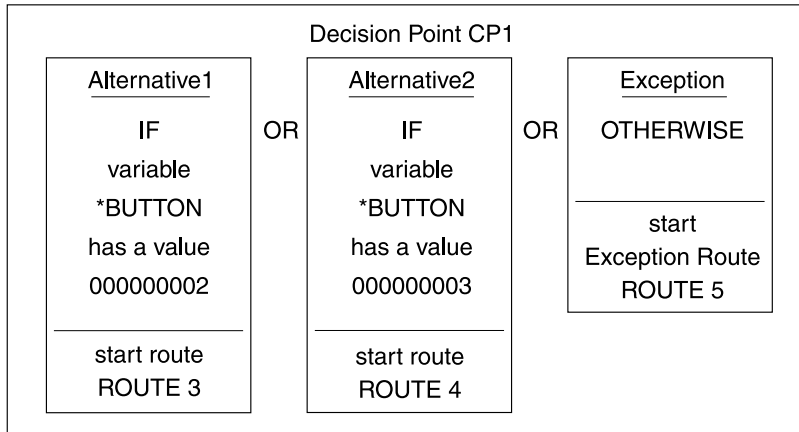


Figure 9. Inside a Decision Point

When the work package arrives at the decision point shown in Figure 9, the work package continues to ROUTE 3 if the variable *BUTTON has a value of 00000002. If the variable *BUTTON has a value of 00000003, then the work package continues to ROUTE 4. If *BUTTON is neither 00000002 nor 00000003, then the work package continues to ROUTE 5.

Advanced Concepts

Before going into greater detail in explaining the workings of decision points, there are a few more work management concepts which need to be understood, as they are crucial to comprehending all the nuances of decision points.

Routes

Take another look at the process diagram you saw earlier in Figure 8 on page 30.

A work package which enters the process goes to the Type 6 workbasket B1. That is the first step in the process. In the second step of the process, the work package goes to the decision point. In the third step in the process, either the work package ends the process or the work package continues to Type 6 workbasket B2. The decision point decides the path flow of the work package. Each arrow in the diagram represents one step. In work management, steps are always recorded sequentially in groups. A group of steps is called a route.

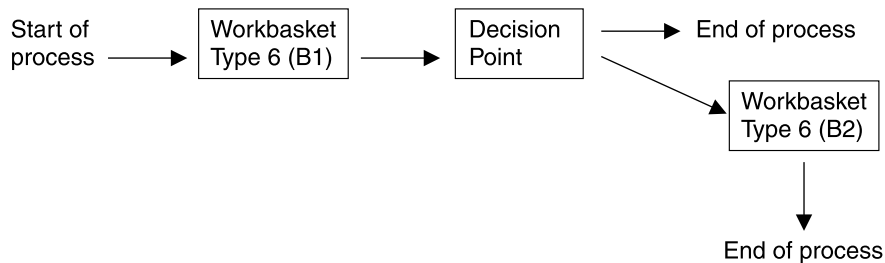
A route consists of one or more steps. When a work package enters a process, it is actually entering the first route of that process. As the work package proceeds through the route, each step in the route is performed using that work package, until the end of the route is reached.

One way that a route can end is with a step which sends the work package to a decision point. This type of step, called a DECIDE command, must always be the last

step in a route, because a work package leaving a decision point always goes to the new route specified in the matched variable/value alternative of the decision point.

There are two route commands that are related to decision points. The first of these is the DECIDE command discussed earlier. When a route processes a DECIDE command step, it sends the work package to the decision point. The other route command is the COPY command. This allows you to assign a value to a variable in the route definition. You can assign a value to the system defined variables *BUTTON and *PRIORITY as well as any customer-defined variables. If you use the COPY command to assign a value to an undefined customer-defined variable, the system creates that variable and assign the specified value.

Now you know enough to understand how the process diagram represents the true structure of a process. The example below shows the process diagram again, this time followed by the actual process objects (routes, a decision point, and two workbaskets, in this case) which make up the process.



represents

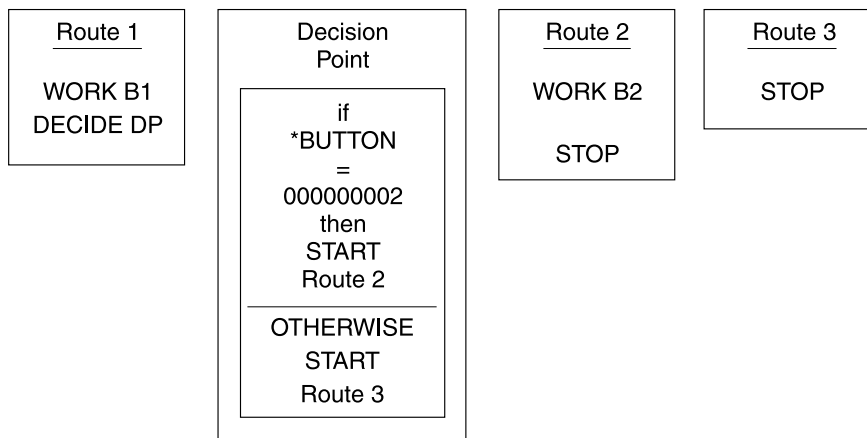


Figure 10. A Very Simple Process with a Decision Point and its Objects

Work Package Instances

The simple process diagram we've been using until now involves a work package moving through the route from start to finish in a linear fashion. But what about the following process?

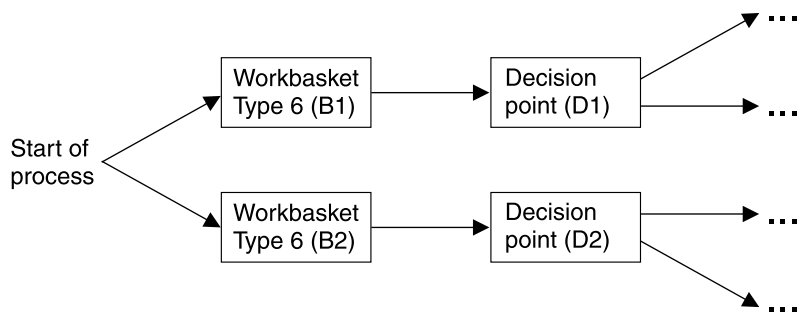


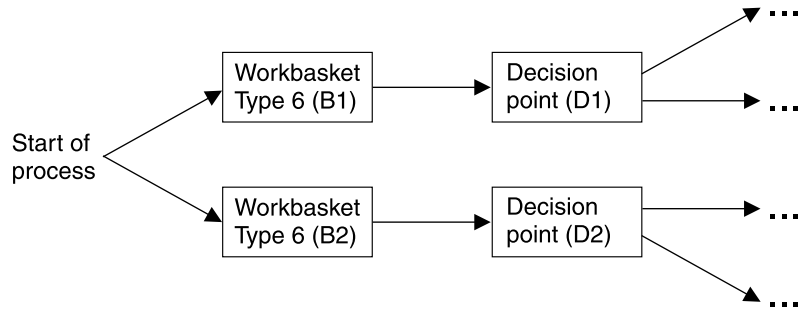
Figure 11. A Parallel Process with Multiple Decision Points

A work package entering this process immediately splits, going to two workbaskets at the same time. Actually, pointers to work packages, not work packages, move through processes. Such pointers contain the names of the work packages they represent. If information about a work package is needed, the system looks at the pointer to find out which work package to examine. These pointers are called instances of a work package. A work package can have up to 99,999 instances, which means that a work package can be in up to 99,999 different places at once during a process.

It is important to note that variables are associated to a specific work package ID and instance. In a completely serial process, there are only one set of variables for a work package. In the above process, each parallel route has its own set of variables and values associated with it, so there are two sets of variables for the work package.

The system defined variables `*BUTTON` and `*PRIORITY` are created when a route or instance is started. The default initial value of `*BUTTON` is 000000000. The default initial value of `*PRIORITY` depends on where the process was started. If the process was started from the reassign function in casework, the priority is the date the case was created. If the process was started from workbasket types 01 through 05, the priority is the scan/index date. If the process was started from file cabinet processing, the priority is the original scan date of the file cabinet document. If the process was started using the BEGWPRCS API, the priority is whatever was passed to the API in the priority parameter field.

When an instance ends, all variables associated with that instance are deleted. A VERY important note is that an instance does not always end when a route ends. The route that is started by a decision point has the same instance as the route that entered the decision point, even though they are two different routes. See Figure 12 on page 35.



represents

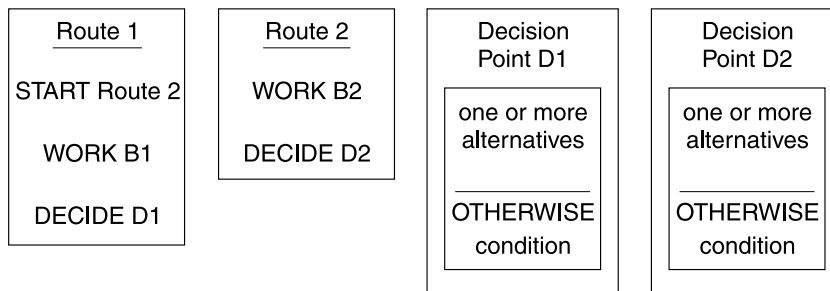


Figure 12. A Parallel Process with Multiple Decision Points and Objects

Note that the workbasket objects were not included, because they don't really add to the example by their presence. As you can see, the second instance of a work package is created by route 1 when the START Route 2 step is performed. START is the only way to create additional instances of a work package. Assuming there are no customer-defined variables, there are 4 variables associated with this process. There is a *BUTTON and *PRIORITY associated with Route 1 and a *BUTTON and *PRIORITY associated with Route 2. In this example we have kept the values of the variables the same, but we could have made the values different in each route.

This should be enough information for you to have a basic understanding of decision points. You should try to define some processes that include decision points and see if your expected results match the actual results.

DECIDE Command

The DECIDE command sends the work package instance to the decision point. When the DECIDE command is processed, the program looks at an alternative defined for this decision point in this process. The system starts the route specified in the matched alternative. Since the alternative matched and a route was started, no more alternatives are checked. There is never more than one route started by a decision point. If there was no match, it tries to compare another alternative defined for this decision point and so on. If all the alternatives are checked and there is no match, the exception route defined in the decision point profile is started.

COPY Command

The COPY route command assigns a value to a work package instance variable. When the system determines that a COPY command should be processed, the variable name and value specified in the COPY command are sent as parameters to the system.

If the COPY command assigns a value to an undefined variable, the COPY command creates that variable with the desired value.

Additional Information

The following can help you better understand how decision points work:

- When a decision point starts a route, variables and corresponding values are identical to the variables and values of the route that went into the decision point.
- Return code EKD-6713 is an informational message that notifies the caller that a variable was created.
- When a collection point is released to the next route or exception route, the values of the variables associated with the route are equivalent to those of the event that activated the collection point. For example, if a route event activates the collection point, the route started by the released collection point has the same variable values as the activating route. If a document event activates the collection point, the route started by the released collection point has variable *BUTTON with a value of 0000000000 and variable *PRIORITY with a value of spaces.
- Variables cannot be deleted from an active work package instance.
- When searching through the alternatives to find a matched variable value, the code reads through the EKD0731 file in reverse order. This means that when you are defining the alternatives in the decision point profile, the last alternative you define is actually the first alternative checked for a match. If a match is found, the other alternatives are not checked.
- The use of customer-defined variables can be a powerful tool. You can use the USEREXIT route command to call your own user exit programs that use the variables with the CHGSYM and QRYSYM APIs.

Here is a simple example of how you could use customer-defined variables: A company processes loans using file cabinet work packages. The requested loan amount might be a file cabinet keyword value. Depending on the loan amount, certain people might be required to approve the loan request. For example, upper management must approve loans over \$500,000, a first-line manager must approve those over \$100,000, and so on. When the person processing the loan request presses continue, a user exit for a custom application is called and determines the loan amount. Depending on the loan amount, the user exit uses the CHGSYM API to create or change a customer-defined variable named LOANRANGE. From the user exit, the process proceeds to a decision point, which routes the work package to the appropriate manager workbasket based on the value of variable LOANRANGE. It might look like this:

```
Route RT1
10 WORK      USERWB
```

```

20 USEREXIT  CUSTCODE
30 DECIDE    DP1
40 STOP
pseudocode
Decision point DP1

Exception route = .....
Alternatives:
If LOANRANGE = 1 then start RT2
If LOANRANGE = 2 then start RT3
If LOANRANGE = 3 then start RT4

Route RT2
10 WORK     UPPERMGMT
20 STOP

Route RT3
10 WORK     MANAGER
20 STOP

Route RT4
10 WORK     ANYBODY
20 STOP

```

Here is the pseudocode for user exit CUSTCODE:

```

If file-cabinet-keyword-amount > 500000
  use CHGSYM to set variable LOANRANGE to 1
else
  if file-cabinet-keyword-amount > 100000
    use CHGSYM to set variable LOANRANGE to 2
  else
    use CHGSYM to set variable LOANRANGE to 3
  end-if
end-if

```

- Variable *BUTTON is set when a function key is pressed from workbasket types 2, 3, and 6.
- *PRIORITY can be used at decision points. This example describes the use of *PRIORITY in a decision point. *PRIORITY can be based on dates. This means that the priority of a document is based on the date it was scanned into the system. If the priority was 970712, it would mean that the document was scanned in on July 12, 1997. Another document scanned into the system later, in August would look like this: 970831 for August 31, 1997. In a decision point profile, you would need to specify the variables you are trying to compare (in this case *PRIORITY). Use the USEREXIT command to call custom code that checks the value of the *PRIORITY variable and sets the variable so that it can be used for the decision point. An example of this is as follows:

Before the DECIDE command in the route profile, insert a USEREXIT command. The user exit code could process as follows:

```
if *PRIORITY < 970801 then set user variable XXXX = 1 otherwise
    set user variable XXXX = 0
```

Then your decision point profile would be as follows:

```
if variable XXXX = 1, start route ROUTE-A
if variable XXXX = 0, start route ROUTE-B
```

When the route reaches the decision point, there is only one *PRIORITY value. If there are two documents in the one work package, the value of *PRIORITY is the scan date of the first document.

- Work package variables are assigned to each work package instance. If a work package is in 2 different workbaskets due to parallel routing, that work package has 2 values for *BUTTON and 2 values for *PRIORITY (as well as any customer-defined variables). An example can be seen in Figure 13.

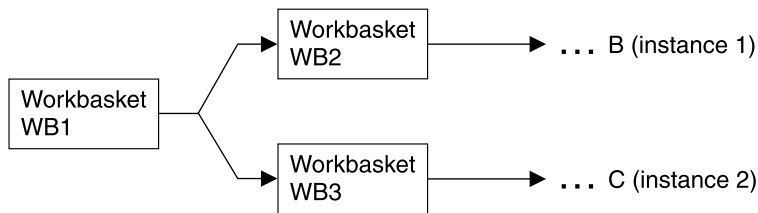


Figure 13. Work Package Variables in a Process

Looking at *BUTTON in this example, at point A, the value of *BUTTON is whatever function key was used to exit workbasket WB1. There is only one value of *BUTTON for this work package since there is only one instance (route). Now the work package is routed in parallel to workbaskets WB2 and WB3. At this point, with WB2 and WB3, there are two values of *BUTTON, one for each active instance. The work management variable file EKD0756 would show two values of *BUTTON for one work package distinguished by instance number. At point B, *BUTTON for instance 1 of the work package contains the function key used to exit WB2. At point C, *BUTTON for instance 2 of the work package contains the function key used to exit WB3.

Chapter 6. Example of a Sequential Routing in a Medical Claim Verification Process

This example describes the processing that occurs with a sequential process. In this example, a department in an insurance company receives medical claim forms in the mail from an insured. The medical claim forms are scanned into the system and then sent through a claims verification process. In the claims verification process, a team of employees reviews the claim to make sure that it is valid. After the medical claim is verified, a software program written by the programmers in the insurance company writes the check for the insured person who submitted the claim. The claim is then filed into the insured's file.

Determining What Workbaskets Are Needed

The analysis begins with the individual logs to determine which workbaskets are needed to support the medical claim process and then uses the group logs to determine the path that the work package follows through that process.

Perform the following steps:

1. **Determine the contents of a work package.** In the medical claim process only one document is used: the medical claim form.
2. **Determine where the document originates.** The insured person sends the document to the insurance company. Users, who work as scanners, scan the medical claim form.
3. **Determine the role each employee or group of employees performs.** Determine the role of each group of employees to identify the workbaskets that you need. Each group of employees has its own in-basket in the traditional manual handling process. We are automating that manual process. The two groups of employees described in the process are as follows:

Scanners and indexers

Open the mail, scan in the medical claim form, and index the medical claim form into Workfolder Application Facility.

Verifiers

Review the medical claim form and send out the check to the insured.

4. **Assign names to the workbaskets.** Use the roles of the employees (in this example, scanners and the verifiers) for the workbasket names. The workbasket names must be 10 characters or less. The workbasket names in this example are as follows:

SCANINDEX

For the employees who scan and index the medical claim form

VERIFY For the employees who verify the claim.

5. **Select the correct type of workbasket.** Based on the activity that the group of employees performs, select the appropriate type of workbasket. You can choose

from six predefined workbaskets or define one of your own. (See Chapter 3, “Workbaskets and Profiles” on page 8 for a complete list of workbasket types, more information about workbasket types, and the rules that apply to each.)

- **Scanning and indexing uses the SCANINDEX workbasket.** In this example, one workbasket must support both scanning and indexing. In Workfolder Application Facility there are several types of workbaskets that support scanning and indexing. There are two distinct types of workbaskets that can completely fill the role as defined in this example:

Workbasket Type 04 - Scan and index into folders

Accepts documents to be scanned into the system, indexes them into folders (cases), and optionally starts them on a work process.

Workbasket Type 05 - Scan and index into file cabinets

Accepts documents to be scanned into the system, indexes them into file cabinets, and optionally starts them on a work process.

Because Workfolder Application Facility in the medical verification department is based on folders, select the Type 4 workbasket to support scanning and indexing into the folder system. If the medical verification department system had been based on file cabinets, you would have selected the Type 5 workbasket. Whether you choose a Type 4 or Type 5 workbasket is based on whether you have set up your Workfolder Application Facility system with file cabinets or folders.

- **Verifying uses the VERIFY workbasket.** For the other workbasket, select the Type 6 workbasket, Work with work packages. As a Type 6 workbasket, the VERIFY workbasket lists the scanned-in medical claim form in individual work packages so the employees can select from a list of work packages as in a queue. Alternatively, the work packages can be presented one at a time so the employees can perform their verification task on the next available work package. In general, for most processes, the Type 6 workbasket is used because the Type 6 workbasket represents work in a queue, similar to a stack of work in an in-basket, waiting to be worked on by your employees.

6. **Determine the security restrictions and tracing activities.** Workbaskets have associated rules. You specify the rules for each workbasket using workbasket profiles at the same time you create the workbasket. Each type of workbasket has a different set of rules that applies to it. (See “Specifying Workbasket Rules and Profiles” on page 9 for more information about workbasket profiles.)

For this example, specify the following workbasket profile information:

- **Scan and index workbasket**
 - The indexers scan before indexing.
 - The work process associated with the workbasket is called MEDICALCHK.
 - The document type is CLAIM. For more information about creating and using document types on your Workfolder Application Facility system, refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: System Administration Guide*.

- The action list for the SCANINDEX workbasket is called SCANACTION.
- The medical claim department is not interested in logging the activities of the employees on each medical claim form.
- The medical claim department does not specify workbasket security.
- **Verify workbasket**
 - The action list for the VERIFY workbasket is called VERICLAIM.
 - The medical claim department is not interested in logging the activities of the employees on each medical claim form.
 - The medical claim department does not specify workbasket security.
 - The work packages in the VERIFY workbasket are presented in a first in, first out (FIFO) order so that items that are received first are processed first.
 - Employees work on the next available work package in the VERIFY workbasket queue instead of selecting a work package from a list.
- **Determine the actions associated with each task your employees perform on a document.** Based on the actions that the two groups of employees perform, assign to the workbasket a list of permissible actions that the employees can perform. This list of permissible actions is called an action list. The action list must include a list of all of the activities that you want your employees to perform on the documents. Each type of workbasket has a different set of permissible options and function keys available to it. (See “Defining Action List Options and Function Keys” on page 11 for more information on action lists.)

For this example, employees perform the following actions:

- **Scan and index workbasket.** The action list for the scan and index workbasket is called SCANACTION. Scan and index type workbaskets do not distinguish the contents of a work package from its document; each work package is a document. Because scan and index type work packages only have a single document, all actions work on the entire contents of the work package. The actions that employees perform are as follows:

Scan the document in

Workfolder Application Facility performs this step.

Enter the index number into panels

Workfolder Application Facility performs this step.

Exit the index panel normally (continue on to the next document to be scanned)

Assign the CONTINUE function to a function key so the employee can simply press the function key and scan in the next document.

Exit the index panel without indexing the document

Assign the EXIT function to a function key so the employee can simply press the function key and exit the function. The user returns to the Workfolder Application Facility menu.

- **Verify workbasket.** The action list for the verify workbasket is called VERICLAIM. As a Type 6 workbasket, the VERIFY workbasket has a long list of possible actions to be performed. However, employees only perform a small subset of the possible actions. Specifically, for each medical claim form in the work package, employees are only allowed to display the medical claim form on the image display using a Display document option.

For each medical claim work package, the employees perform a set of actions against the entire set of documents in the work package (even though the medical claim package only has one document, it is important to keep the distinction between documents and work packages in mind). The actions that the employees should perform with the entire work package are as follows:

- Make comments on the contents of the medical claim form and store the comments with the medical claim form using the Comments key.
- Look at the entire contents of the work package by sending the contents to the image display terminal using the Display folder function. Because there is only one document in the work package, this is identical to the Display document option.
- Exit this work package without processing it. Exit this panel, and be able either to select another work package or to select another workbasket using the Cancel function. Because the VERIFY workbasket was defined not to produce a listing of work packages, the Cancel function returns users to a position that allows them to select an alternative workbasket or to work on the next available work package in the current workbasket queue.
- Exit the work package and exit the function. Assign the Exit function to a function key that allows the employee to simply press the function key and exit the function. The user returns to the Workfolder Application Facility menu. (Contrast with the Cancel function key defined above.)
- Issue the check and continue on to the next medical claim form using the Customer-defined key. You must program the action of the customer-defined key yourself to send the name, address, and amount of the claim to your own check writing program. When you write the program, different return code values from your program determine how the work process proceeds. A return code of 1 tells the system to perform the action of Continue so that the employee can continue on and process the next medical claim form. Refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: Programming Interfaces Guide* for more information about how to write a program to use the appropriate return codes.

Finally, assign the name Issue check to the customer-defined key function so employees issue a check when they have finished verifying the claim.

Identifying the Flow of Work Across the Process

Use the information from the collection of logs to identify the flow of work across the entire medical department process. Follow these steps.

1. **Determine the final product of the work process in your department.** The final product of the medical verification process is a check that is sent to the insured. The process in this example is called MEDICALCHK to represent the fact that the department issues medical checks.
2. **Determine the path a work package takes from beginning to end.** In the medical verification process, the medical claim form is scanned by the scanners and indexers. The work package containing the medical claim form is then sent to the verifiers. After the claim is verified, a check for the amount of the claim is sent to the insured. After the check is issued, the claim is filed in the insured's folder.
3. **Determine the places in your work process where a work package must wait.** The work package does not wait for any other document or process.
4. **Determine the maximum time of each step in your work process.** Because the work package does not wait, you can assign any number of employees to your two workbaskets, and you do not have any other time criteria.
5. **Determine any special handling or routing decisions.** Because the medical claim form is only handled by two groups, there are no special handling rules or decisions.
6. **Determine whether an action is manual or is handled by a software program.** The major actions performed in the medical claim department are as follows:

Scan and index	This step is manual.
Verify	This step is manual.
Write the check	This step requires the use of function key Exit for workbasket Type 6 (user exit number 38, program EKDWB06F03).
7. **Determine where the contents of the work packages go after the work process has completed.** The medical claim form is automatically filed in the insured's Workfolder Application Facility folder.

Defining the Medical Verification Process

With the information outlined in the previous sections, it is now possible to completely define the Medical Verification process described in this example. There are two workbaskets (SCANINDEX and VERIFY), each with their own action lists (SCANACTION and VERICLAIM). The definitions of these four objects are shown in the following sections. In addition, the following sections contain the complete routing

process depicted using both routes and commands using Workfolder Application Facility 5250 panels, and paths using Work Management Builder symbols.

If you intend to test the examples using either Work Management Builder or 5250 panels to create the action lists before you create the workbaskets. One workbasket variable specified in the workbasket profile is the default action list. You must create an action list name before you can use it in a workbasket.

Workbaskets and Profiles Used in the Medical Claim Verification Process

The complete details of the workbaskets and profiles used in the Medical Claim Verification process are defined in the following sections.

SCANINDEX Workbasket Details

Figure 14 on page 45 shows the details needed to define the SCANINDEX workbasket. Note the following items:

- The security level is set to 0 because there are no security considerations defined for this process.
- Neither history nor user activity is logged. This also means that no information is necessary about the Report Key fields since there is no logging.
- The workbasket type is a Type 4, Folder scan and index.
- The work process that starts as a result of the scanning is MEDICALCHK.
- When a document of type CLAIM is scanned into the system, a work package is created containing that medical claim form. The work package is routed along the process defined in MEDICALCHK.
- The action list associated with this workbasket is called SCANACTION. It must be defined as a Type 1 - Index action list.
- Work packages are not listed at this workbasket because the work packages consist of individual documents waiting to be scanned and indexed.
- The work order is not applicable because the work order is determined by the stack of documents (medical claim forms) waiting to be scanned.
- In this example, documents must be scanned before they are indexed.
- Since this is a scan and index type workbasket, reassigning work packages to this workbasket does not apply. The work is the stack of documents waiting to be scanned.

Object type	: Workbasket
Name	: SCANINDEX
Description	: Scan and index into folders
Security class	: 0
History log level	: No logging
User activity log level	: No logging
Report key	:
	:
	:
Type	: Folder scan and index
Work process	: MEDICALCHK
File cabinet	: N/A
Document type	: CLAIM
Action list	: SCANACTION
List work package	: N/A
Work order	: N/A
Scan before index	: Yes
Allow reassign to	: No

Figure 14. Details of the SCANINDEX Workbasket

VERIFY Workbasket Details

Figure 15 on page 46 shows the details needed to define the VERIFY workbasket.

Note the following items:

- The security level is set to 0 because there are no security considerations defined for this process.
- Neither history nor user activity is logged. This also means that no information is necessary about the Report Key fields since there is no logging.
- The workbasket type is a general Type 6, Work with work packages.
- The action list associated with this workbasket is called VERICLAIM. It must be defined as a Type 2 - Work action list.
- As a default, work packages are not listed at this workbasket. In general, this prevents users from selecting which work package they work on next. However, an individual user profile can be set so that the individual user can override that default value and thereby list the work packages.
- The work order is first in, first out (FIFO). This means that the work packages are presented to the user in the order in which the work packages arrive at the workbasket.
- Since this is not a scan and index type workbasket, scan before indexing is not an option.
- Since there are only two workbaskets in this process, reassigning work packages to this workbasket does not make sense. There is no other workbasket that could assign work to it. Work packages only arrive from the scan and index workbasket.

Object type	: Workbasket
Name	: VERIFY
Description	: Verify claim, issue check
Security class	: 0
History log level	: No logging
User activity log level	: No logging
Report key	:
	:
	:
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: VERICLAIM
List work package	: No
Work order	: First in first out
Scan before index	: No
Allow reassign to	: No

Figure 15. Details of the VERIFY Workbasket

Action List Definitions Used in the Medical Claim Verification Process

SCANACTION Action List Details

Figure 16 shows the details needed to define the SCANACTION action list. Note the following items:

- The action list is defined as a Type 1 - Index action list.
- There are only two actions that users can perform:

Next Lets the work package continue on to the next route in the work process.

Exit Lets users exit the entire operation of scanning and indexing, and creating work packages.

Workfolder Application Facility supports all other actions.

Object type	: Action List
Name	: SCANACTION
Description	: Scan and index into folders
Type	: Index
List of functions	:
Description	Number Prompt
-----	-----
Continue key	F2 Next
Exit key	F3 Exit

Figure 16. Details of the SCANACTION Action List

VERICLAIM Action List Details

Figure 17 shows the details needed to define the VERICLAIM action list. Note the following items:

- The action list is defined as a Type 2 - Work action list.
- The only actions that users can perform are as follows:
 - Display** To display the medical claim form on the image display.
 - Exit** To exit the work package. The work package remains in the workbasket to be worked later.
 - Comments** To allow users to make comments about the medical claim form.
 - Cancel** To leave the current work package, but not the function.
 - Issue check** To issue a check. This is a customer-defined key. As stated previously, this function allows users to send information about the work package to a line-of-business application, and then files the work package.

Object type	: Action List	
Name	: VERICLAIM	
Description	: Claim Verification	
Type	: Work	
List of functions	:	
Description	Number	Prompt
Display document	5	Display
Exit key	F3	Exit
Comments key	F11	Comments
Cancel key	F12	Cancel
Customer-defined key	F15	Issue check

Figure 17. Details of the VERICLAIM Action List

Work Process Definition for Medical Claim Verification

Figure 18 on page 48 shows the details needed to define the work process for the medical claim verification example. Note the following items:

- The work process MEDICALCHK is started whenever a document type of CLAIM is scanned. Scanning automatically creates a work package and begins the routing of the work package based on the information in SCANINDEX workbasket profile.
- The WORK command is associated with the workbasket VERIFY (parameter 1). Specify parameter 2 only if you want to override the default action list associated with the WORK command. In this example, the default action list VERICLAIM is used, so parameter 2 is not specified.

- The STOP command ends the process and places the work package in an associated Workfolder Application Facility folder.

Object type	: Work Process		
Name	: MEDICALCHK		
Description	: Medical verification process		
Starting route name	: VER		
List of routes	:		
Route name	: VER		
Step	Command	Parm 1	Parm 2
10	WORK	VERIFY	
20	STOP		

Figure 18. Details of the Routes and Commands Used in the Medical Claim Verification Process

Finally, Figure 19 shows the medical verification process graphically. Note that with Work Management Builder, the work node symbol represents both the workbasket and the action list for the Type 6 workbasket VERIFY.

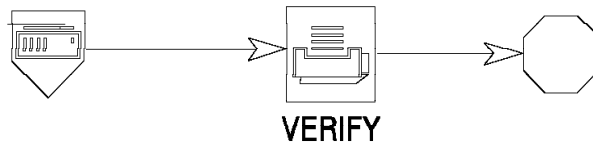


Figure 19. Completed Medical Verification Process

If you are using Work Management Builder to generate Workfolder Application Facility database objects, follow these steps.

- Verify the process
- Check in both of the workbaskets
- Check in both of the action lists
- Check in the work process itself

For more information about Work Management Builder, refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: User's Guide for the Work Management Builder*.

Chapter 7. Example of Multiple Routing in a Medical Claim Verification Process

The previous chapter described a simple sequential process: a medical claim form was scanned, a group of employees verified the medical claim, sent out a check for the claim amount, and filed the claim in the insured's folder.

This chapter describes a more complex process. In this example, the medical claim form goes to a group of employees, who perform an initial review of the claim to be sure that the insured has completed all claim information:

- If the medical claim form is complete and it is over \$10,000, the medical claim form goes to a team who reviews the charges on the claim to be sure that the charges are reasonable.
- If the medical claim form is complete and it is under \$10,000, the medical claim form goes to a different team, who reviews the charges to be sure that the charges are reasonable.
- If the medical claim form is incomplete, it goes to another team, who contacts the insured to obtain the remaining information.

After the initial review, the medical claim form goes to one of these teams:

Over \$10,000 team

This team reviews the charges. If the charges are reasonable, the medical claim form is sent to a final verification team. If the charges are not reasonable, a letter is sent to the insured and the claim is filed in the insured's folder.

Under \$10,000 team

This team reviews the charges. If the charges are reasonable, the medical claim form is sent to a final verification team. If the charges are not reasonable, a letter is sent to the insured and the claim is filed in the insured's folder.

Completeness review team

This team contacts the insured to get the missing information. After the team receives the information, the work package is sent back to the initial review team.

Determining What Workbaskets Are Needed

The analysis begins with the individual logs to determine which workbaskets are needed to support the medical claim process and then uses the group logs to determine the path that the work package follows through that process.

1. **Determine the contents of a work package.** This is identical to the previous example. The content of the work package is a medical claim form.
2. **Determine where the document originates.** This is identical to the previous example. The insured sends the document to the insurance company. The users, who are scanners, scan the medical claim form.
3. **Determine the role each employee or group of employees performs.**

Scanners and indexers

Open the mail, scan in the medical claim form, and index the medical claim form into Workfolder Application Facility.

Initial reviewers

Decide whether or not the claim is complete, and forward the claim to one of the three review teams.

Under 10,000 reviewers

Review the claims under \$10,000 to ensure that the claims are reasonable and forward the claim to the final verification team.

Over 10,000 reviewers

Review the claims over \$10,000 to ensure that the claims are reasonable and forward the claim to the final verification team.

Completeness reviewers

Contact the insured to get the information that was omitted from the claim and send the claim back to the initial review team after they receive all of the information from the insured.

Verifiers

Perform the final review of the medical claim form and send the check to the insured.

4. **Assign names to the workbaskets.** Use the roles of the employees for the workbasket names. The workbasket names must be ten characters or less. The workbasket names in this example areas follows:
 - SCANINDEX2 for the employees who scan and index the medical claim form
 - REVIEW for the employees who perform the initial review
 - OVER10 for the employees who review claims over \$10,000
 - UNDER10 for the employees who review claims under \$10,000
 - CONTACT for the employees who contact the insured for more information
 - VERIFY for the employees who review the previous work and issue a check.
5. **Select the correct type of workbasket.** Based on the activity that the group of employees perform, select the appropriate type of workbasket. (See Chapter 3, "Workbaskets and Profiles" on page 8 for a complete list of workbasket types, more information about workbasket types, and the rules that apply to each.)
 - For scanning and indexing, use a Type 4 workbasket similar to the one used in the previous example.
 - For all other workbaskets use the general Type 6 workbasket, Work with work packages.
6. **Determine the security restrictions and tracing activities.** Workbaskets have associated rules. You specify the rules for each workbasket using workbasket profiles at the same time you create the workbasket. (See "Specifying Workbasket Rules and Profiles" on page 9 for more information about workbasket profiles.)

For this example, the profile information that you need to specify is as follows:

- **Scan and index workbasket**

- The indexers scans before indexing.
- The work process associated with the workbasket is called MEDVER2. Note that the name of this work process is different from the name used in the previous example. If you specified the same work process name for both workbaskets then scanning and indexing using either workbasket would initiate the same process.
- The document type is CLAIM.
- The action list for the SCANINDEX2 workbasket is called SCANACTION.
- The medical claim department is not interested in logging the activities of the employees on each medical claim form.
- The medical claim department does not specify workbasket security.
- **Review workbasket**
 - The action list for the REVIEW workbasket is called REVIEW.
 - The medical claim department is not interested in logging the activities of the employees on each medical claim form.
 - The medical claim department does not specify workbasket security.
 - The work packages in the REVIEW workbasket are presented in a priority order so that items that are received with the highest priority are processed first. You must set up a priority queuing for this workbasket because it contains claims from today (those just scanned in) and those that are sent back to the REVIEW workbasket from the completeness review team. Claims with the highest priority are reviewed first.

In this example, use the default priority of case creation date for the priority value of the work package. This means that the older work packages are listed first (those sent back to REVIEW).
 - Employees work on the next available work package in the REVIEW workbasket queue based on priority instead of selecting a work package from a list.
 - Specify that work can be reassigned to the REVIEW workbasket. Work packages from the workbasket CONTACT are sent back to the REVIEW workbasket after the insured has been contacted.
- **Over10 workbasket**
 - The action list for the OVER10 workbasket is called APPRVRJT because the employees are approving and rejecting claims.
 - The medical claim department wants to log all activity associated with the expensive claims. Since the employees work with claims over \$10,000, specify that logs are kept to record both the access to the workbasket and the actions performed on the work packages.

History log level

Specify a log level of 1 to log the number of times the work package is assigned to the workbasket.

User activity log level

Specify a log level of 1 because you want to log whether the work package has been accessed.

- The medical claim department also specifies the term OVER10 in the Report key field. Although you can specify terms for up to three Report key fields, one term is sufficient to locate the log information in this example.
- The medical claim department specifies a workbasket security level of 300. If a user's security range does not include this level of security, the user is not able to access this workbasket. A user's workbasket security range is defined in the user profile. Refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: System Administration Guide* for more information about user profiles.
- The work packages in the OVER10 workbasket are presented in a first in, first out (FIFO) order so that items that are received first are processed first.
- Employees work on the next available work package in the OVER10 workbasket queue instead of selecting a work package from a list.
- Specify that work cannot be reassigned to the OVER10 workbasket.
- **Under10 workbasket**
 - The action list for the UNDER10 workbasket is called APPRVRJT because the employees are approving and rejecting claims. Note that this is the same action list used in the OVER10 workbasket.
 - The medical claim department is not interested in logging the activities of the employees on each medical claim form for the claims that are under \$10,000.
 - The medical claim department does not specify workbasket security.
 - The work packages in the UNDER10 workbasket are presented in first in first out order so that items that are received first is processed first.
 - Employees work on the next available work package in the UNDER10 workbasket queue instead of selecting a work package from a list.
 - Specify that work cannot be reassigned to the UNDER10 workbasket.
- **Contact workbasket**
 - The action list for the CONTACT workbasket is called CONTACT.
 - The medical claim department wants to log all activity associated with the work packages that require the insured be contacted so the company can track the activity. Specify that logs be kept to record both the access to the workbasket and the actions performed on the work packages.

History log level

Specify a log level of 3 because you want to log all activity related to the work package.

User activity log level

Specify a log level of 2 because you want the activities to be logged before the employee leaves the workbasket.

- The medical claim department also specifies the term CONTACT in the Report key field. Although you can specify terms for up to three Report key fields, one term is sufficient to locate the log information in this example.
- The medical claim department does not specify workbasket security.
- The work packages in the CONTACT workbasket are presented in a priority order so that items that are received with the highest priority are processed first. You must set up a priority queuing for this workbasket because it contains claims from today (those from the REVIEW team) and those for which have not been able to contact the insured. Claims with the highest priority are reviewed first.

In this example, use the default priority of case creation date for the priority value of the work package. This means that the older work packages are listed first.

- Employees can select work packages in the CONTACT workbasket so that, if an insured calls an employee, the employee can select that specific work package.
- Specify that work can be reassigned to the CONTACT workbasket so that any claim that has missing information can be reassigned to the workbasket.

- **Verify workbasket**

- The action list for the VERIFY workbasket is called VERICLAIM. This is the identical action list used in the previous example.
- The medical claim department is not interested in logging the activities of the employees on each medical claim form.
- The medical claim department does not specify workbasket security.
- The work packages in the VERIFY workbasket are presented in a first in, first out (FIFO) order so that items that are received first are processed first.
- Employees work on the next available work package in the workbasket queue instead of selecting a work package from a list.

7. **Determine the actions associated with each task your employees perform on a document.** The actions that the employees perform are as follows.

- **Scan and index workbasket.** The action list for the scan and index workbasket is called SCANACTION. You can use the same action list that was used in the previous example.
- **The Type 6 Workbaskets.** Type 6 workbaskets let you select from a long list of possible actions. Many of those actions are common across the

workbaskets defined in this example. Specifically, for each medical claim form in the work package, the employees do the following.

- **Display the medical claim form on the image display using a Display document option.** Because this work package contains a single document, the document is displayed automatically on the image display. In Workfolder Application Facility, single documents are always displayed automatically. If the work package has more than one document, none are displayed automatically, the user must explicitly select a document to be displayed. You specify the Display document option here because more documents might be added to the work package later and the automatic display would not operate. You might then wonder why the documents could not be displayed.
- **Display the document details using a Document details option.**

For each medical claim work package, employees perform a set of actions against the entire set of documents in the work package (even though the medical claims package only has one document, it is important to keep the distinction between documents and work packages in mind). The actions that the employees should perform with the entire work package are as follows.

- Make comments on the contents of the medical claim form and store the comments with the medical claim form using the Comments key.
- Look at the entire contents of the work package by sending the contents to the image display terminal using a Display folder function key. Because the work package contains only one document, this is identical to the Display document option.
- Exit this work package without processing its panel. Exit this panel, and continue with some other activity not associated with the medical claim process, using the Exit key.
- Additional actions based on the individual workbaskets
 - **REVIEW workbasket.** For the REVIEW workbasket, the employees route the work package to one of three places (OVER10, UNDER10, or CONTACT). Employees accomplish this task by using the Continue function. (See “Using the Action List Functions to Make Routing Decisions” on page 16 for an extended discussion.)
 - Continue to the OVER10 team using the F5=Over10 key
 - Continue to the UNDER10 team using the F2=Under10 key
 - Continue to the CONTACT team using the F8=Followup key. See Chapter 5, “Understanding Decision Points” on page 30 for more information about the use of the function key variable, *BUTTON, in decision points and exception routing.

The action list is called REVIEW.

- **OVER10 workbasket.** The employees at the OVER10 workbasket decide whether or not the amount claimed is valid. They send the work package on to the VERIFY team if the claim is valid using the

F5=Valid function key. The function assigned to the F5=Valid function key must be the continue function. If the claim is not valid, employees send the work package on to the DEFAULT user exit, together with a reason code so that the insured receives a letter explaining why the insured is not receiving a check. The continue function is also assigned to the F6=Reject function key. This action list is called APPRVRJT for approve or reject.

- **UNDER10 workbasket.** The employees at the UNDER10 workbasket decide whether or not the amount claimed is valid. They send the work package on to the VERIFY team if the claim is valid using the F5=Valid function key. If the claim is not valid, employees send the work package on to the DEFAULT user exit, together with a reason code so that the insured receives a letter explaining why the insured is not receiving a check. The continue function is also assigned to the F6=Reject function key. This action list is called APPRVRJT for approve or reject. Note that this action list is identical to that used by the OVER10 workbasket. You can use the same action list because the actions are identical, and the decision point decides exactly where the work package continues on to.
- **CONTACT workbasket.** The employees at the CONTACT workbasket contact the insured, get the missing information from the medical claim form, and then send the work package containing the medical claim form back to the REVIEW workbasket. In addition to the general actions defined, these users can also view the history of the work package using an F19=View History function key. In addition, create an assignment of the Continue function so that F5=Complete. Users can then press F5 when they finish gathering information from the insured. F5 lets the work package continue on in the process to the next workbasket. Note that for this workbasket, the next workbasket is actually the originating workbasket (REVIEW). This action list is called CONTACT.
- **VERIFY workbasket.** The action list for the VERIFY workbasket is called VERICLAIM. You can use the same action list that was used in the previous example.

Identifying the Flow of Work Across the Process

Use the information from the collection of logs to identify the flow of work across the entire medical department process.

1. **Determine the final product of the work process in your department.** The final product of the medical verification process is a check that is sent to the insured. Call this process MEDVER2 to represent the fact that this is the second of the Medical Verification processes.
2. **Determine the path a work package takes from beginning to end.**
 - In the medical verification process, the medical claim form is scanned by the scanners and indexers.

- The work package containing the medical claim form is then sent to the reviewers.
- After the claim is reviewed, it is sent to one of these places:

OVER10 team

From the OVER10 team, the work package is sent to either the DEFAULT user exit or to the VERIFY team.

UNDER10 team

From the UNDER10 team, the work package is sent to either the DEFAULT user exit or to the VERIFY team.

CONTACT team

From the CONTACT team, the work package is sent to the REVIEW team.

- From either the OVER10 or UNDER10 teams, the work package containing the medical claim form is then sent to the VERIFY team. After the medical claim form is verified, a check for the amount of the claim is sent to the insured. After the check is issued, the claim is filed in the insured's folder.

3. **Determine the places in your work process where a work package must wait.**
The work package does not wait for any additional information to be added to the work package.
4. **Determine the maximum length of time of each step in your work process.**
Because the work package does not wait, you can assign any number of employees to the two workbaskets and you do not have any other time criteria.
5. **Determine whether an action is manual or is handled by a software program.**
This is identical to the previous example.
6. **Determine where the contents of the work packages go after the work process is completed.** The medical claim form is filed automatically in the insured's Workfolder Application Facility folder at the end of the process.

Defining the Medical Claim Verification Process

With the information outlined in the previous sections, it is now possible to completely define the second medical verification process.

Workbasket Definitions Used in the Medical Claim Verification Process

The complete details of the workbaskets and profiles used in the Medical Claim Verification process are defined in the following sections.

SCANINDEX2 Workbasket Details

Figure 20 on page 57 shows the details needed to define the SCANINDEX2 workbasket. Note the following items:

- The security level is set to 0 because there are no security considerations defined for this process.

- Neither history nor user activity is logged. This also means that no information is necessary about the Report Key fields since there is no logging.
- The workbasket type is a Type 4, Folder scan and index.
- The work process that starts as a result of the scanning is MEDVER2.
- When a document of type CLAIM is scanned into the system, a work package is created containing that medical claim form. The work package is routed along the process defined in MEDVER2.
- The action list associated with this workbasket is called SCANACTION. It must be defined as a Type 1 - Index action list.
- Work packages are not listed at this workbasket because the work packages consist of individual documents waiting to be scanned and indexed.
- The work order is not defined because the work order is determined by the stack of documents (medical claim forms) waiting to be scanned.
- In this example, documents must be scanned before they are indexed.
- Since this is a scan and index type workbasket, reassigning work packages to this workbasket does not apply. The work is the stack of documents waiting to be scanned.

Object type	: Workbasket
Name	: SCANINDEX2
Description	: Scan and index into folders
Security class	: 0
History log level	: No logging
User activity log level	: No logging
Report key	:
	:
	:
Type	: Folder scan and index
Work process	: MEDVER2
File cabinet	:
Document type	: CLAIM
Action list	: SCANACTION
List work package	: No
Work order	: Undefined
Scan before index	: Yes
Allow reassign to	: No

Figure 20. Details of the SCANINDEX2 Workbasket

REVIEW Workbasket Details

Figure 21 on page 58 shows the details needed to define the REVIEW workbasket. Note the following items:

- The workbasket type is Type 6, Work with work packages.

- No security is specified.
- Neither history nor user activity is logged. This also means that the Report key fields do not apply.
- The action list is also called REVIEW.
- Work packages are not listed. Users work on the next available work package without being able to select from a list of work packages.
- The work packages are worked in priority order based on the date that the document in the work package is scanned.
- Work packages can be reassigned to this workbasket.

Object type	: Workbasket
Name	: REVIEW
Description	: Initial review
Security class	: 0
History log level	: No logging
User activity log level	: No logging
Report key	:
	:
	:
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: REVIEW
List work package	: No
Work order	: Priority
Scan before index	: No
Allow reassign to	: Yes

Figure 21. Details of the REVIEW Workbasket

OVER10 Workbasket Details

Figure 22 on page 59 shows the details needed to define the OVER10 workbasket. Note the following items:

- The workbasket is a Type 6, Work with work packages workbasket.
- The security level is set to 300. If a user's security range does not include this level of security, the user is not able to access this workbasket. A user's workbasket security range is defined in the user profile.
- Both the history and user activity is logged for work package access. The log information can be accessed using the Over10 Report key field.
- The action list is called APPRVRJT. It is the same action list used for the UNDER10 workbasket.

- Work packages are not listed. Users work on the next available work package without being able to select from a list of work packages.
- The work packages are worked in first in, first out (FIFO) order. The work packages that arrive first in the workbasket are worked first.
- Work packages cannot be reassigned to this workbasket.

Object type	: Workbasket
Name	: OVER10
Description	: Claims over \$10,000
Security class	: 300
History log level	: Work package access
User activity log level	: Work basket access
Report key	: OVER10
	:
	:
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: APPRVRJT
List work package	: No
Work order	: First in first out
Scan before index	: No
Allow reassign to	: No

Figure 22. Details of the OVER10 Workbasket

UNDER10 Workbasket Details

Figure 23 on page 60 shows the details needed to define the UNDER10 workbasket. Items to note are that the workbasket details are identical to the OVER10 workbasket except that no security is specified and no logging is performed.

Object type	: Workbasket
Name	: UNDER10
Description	: Claims under \$10,000
Security class	: 0
History log level	: No logging
User activity log level	: No logging
Report key	:
	:
	:
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: APPRVRJT
List work package	: No
Work order	: First in first out
Scan before index	: No
Allow reassign to	: No

Figure 23. Details of the UNDER10 Workbasket

CONTACT Workbasket Details

Figure 24 on page 61 shows the details needed to define the CONTACT workbasket.

Note the following items:

- The workbasket is a Type 6, Work with work packages workbasket.
- The security level is 0.
- The most that can be logged about both history and user activity is logged. All actions performed from this workbasket are logged. The information in the logs can be accessed using the Contact key word.
- The action list is also called CONTACT.
- Because the users of this workbasket receive telephone calls from the insured, the users must be able to select individual work packages to work on. The work packages are shown in a list so that users can select which one they want to work with.
- The work packages are listed in priority order so that follow up with the insured can also be based on priority.
- Since the CONTACT workbasket contains work packages that are incomplete, work packages can be reassigned to the workbasket for more follow up.

Object type	: Workbasket
Name	: CONTACT
Description	: Contact the insured
Security class	: 0
History log level	: Level 2 and workbasket option codes
User activity log level	: Level 3
Report key	: CONTACT
	:
	:
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: CONTACT
List work package	: Yes
Work order	: Priority
Scan before index	: No
Allow reassign to	: Yes

Figure 24. Details of the CONTACT Workbasket

VERIFY Workbasket Details

For the details of the VERIFY workbasket, see Figure 15 on page 46.

Action List Definitions Used in the Medical Claim Verification Process

The complete details of the action lists used in the Medical Claim Verification process are defined in the following sections.

VERICLAIM Action List Details

The VERICLAIM action list was defined in the previous example. See Figure 17 on page 47 for the details needed to define the VERICLAIM action list.

REVIEW Action List Details

Figure 25 on page 62 shows the details needed to define the REVIEW action list. Note the following items:

- The action list is a Type 2 - Work action list.
- There are three Continue functions defined. This lets users make the routing choices based on meaningful information:
 - If the users press F2, the work package continues on to the UNDER10 workbasket.
 - If the users press F5, the work package continues on to the OVER10 workbasket.
 - If the users press F8, the work package continues on to the CONTACT workbasket (using the function key labelled Followup).

Note that the users do not have to know a user ID or even a specific route to make the decisions. The names of the prompt fields reflect the action to take.

For more information about the meaning of the other specific functions and options, see “Defining Action List Options and Function Keys” on page 11.

Object type	: Action List	
Name	: REVIEW	
Description	: Review claim, route based on \$	
Type	: Work	
List of functions	:	
Description	Number	Prompt
Display document	5	Display
Document details	8	Details
Continue key	F2	Under10
Exit key	F3	Exit
Display folder	F4	Display
Continue key	F5	Over10
Continue key	F8	Followup
Comments key	F11	Comments

Figure 25. Details of the REVIEW Action List

APPRVRJT Action List Details

Figure 26 on page 63 shows the details needed to define the APPRVRJT action list.

Note the following items:

- The action list is a Type 2 - Work action list.
- The decision made by the users is to press F5=Valid if the claim is valid. If the F5 key is pressed, the work package continues on and a check is issued.

For more information about the meaning of the other specific functions and options, see “Defining Action List Options and Function Keys” on page 11.

Object type	: Action List	
Name	: APPRVRJT	
Description	: Approve or reject	
Type	: Work	
List of functions	:	
Description	Number	Prompt
<hr/>	<hr/>	<hr/>
Display document	5	Display
Document details	8	Details
Exit key	F3	Exit
Display folder	F4	Display
Continue key	F5	Valid
Continue key	F6	Reject
Comments key	F11	Comments

Figure 26. Details of the APPRVRJT Action List

CONTACT Action List Details

Figure 27 shows the details needed to define the CONTACT action list. Note the following items:

- The action list is a Type 2 - Work action list.
- The F5 key is assigned to the Continue function. It is labelled Complete because that is the judgement that the users are making. When F5 is pressed, the work package continues on to the initial REVIEW workbasket from which it originated.
- Because there could be a lot of work related to this specific work package, F19=History is assigned to the workbasket. This lets users look at the history of the work package.

For more information about the meaning of the other specific functions and options, see “Defining Action List Options and Function Keys” on page 11.

Object type	: Action List	
Name	: CONTACT	
Description	: Contact customer	
Type	: Work	
List of functions	:	
Description	Number	Prompt
<hr/>	<hr/>	<hr/>
Display document	5	Display
Document details	8	Details
Exit key	F3	Exit
Display folder	F4	Display
Continue key	F5	Complete
Comments key	F11	Comments
View history key	F19	History

Figure 27. Details of the CONTACT Action List

Work Process Definition for Medical Claim Verification

Figure 28 shows the details needed to define the work process for the medical claim verification example.

Object type	: Work Process		
Name	: MEDVER2		
Description	: Medical verification example 2		
Starting route name	: 20		
Process item version	: 0		
List of routes	:		
Route name	: 20		
Step	Command	Parm 1	Parm 2
10	WORK	REVIEW	
20	DECIDE	UNDER/OVER	
Route name	: 30		
Step	Command	Parm 1	Parm 2
10	WORK	UNDER10	
20	DECIDE	FINAL REV	
Route name	: 40		
Step	Command	Parm 1	Parm 2
10	WORK	VERIFY	
20	STOP		
Route name	: 50		
Step	Command	Parm 1	Parm 2
10	USEREXIT		
20	STOP		

Figure 28 (Part 1 of 3). Details of the Routes and Commands Used in the Medical Claim Verification Process (Multiple Routing)

Route name			: 60
Step	Command	Parm 1	Parm 2
10	START	50	
Route name			: 70
Step	Command	Parm 1	Parm 2
10	WORK	OVER10	
20	DECIDE	FINAL REV	
Route name			: 80
Step	Command	Parm 1	Parm 2
10	WORK	CONTACT	
20	START	20	
List of decision points			:
Decision point name			: FINAL REV
Description			: Send check or send rejection
Decision branch ID			: 1
Completion route name			: 50
Decision branch trigger			: Exception condition
Decision branch ID			: 2
Completion route name			: 40
Decision branch trigger			: *BUTTON EQ 0000000005
List of decision points			:
Decision point name			: UNDER/OVER
Description			: Under or over \$10,000
Decision branch ID			: 1
Completion route name			: 80
Decision branch trigger			: Exception condition

Figure 28 (Part 2 of 3). Details of the Routes and Commands Used in the Medical Claim Verification Process (Multiple Routing)

Decision branch ID	: 2
Completion route name	: 30
Decision branch trigger	: *BUTTON EQ 000000002
Decision branch ID	: 3
Completion route name	: 70
Decision branch trigger	: *BUTTON EQ 000000005

Figure 28 (Part 3 of 3). Details of the Routes and Commands Used in the Medical Claim Verification Process (Multiple Routing)

Finally, Figure 29 on page 67 shows the second medical claim verification process graphically. For more information about Work Management Builder, refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: User's Guide for the Work Management Builder*.

Even though the employee roles are represented by individual workbaskets, you can authorize the same employee to perform work in multiple workbaskets. Those employees might have different user IDs and user profiles. For example, you might want the user assigned to the REVIEW workbasket to review work in the morning. In the afternoon, you might authorize the same user to work with the work packages in the CONTACT workbasket. In fact, because the number of work packages in the CONTACT workbasket probably grow quickly because the user must contact the insured, you can also authorize the employees assigned to the OVER10 and UNDER10 workbaskets to work on the work packages assigned to the CONTACT workbasket. For more information about working with user profiles, refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: System Administration Guide*.

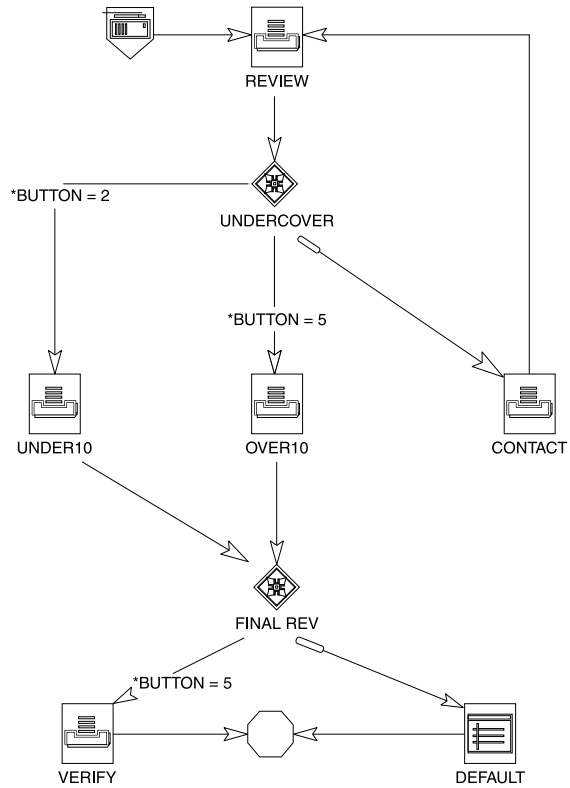


Figure 29. Completed Medical Claim Verification Process (Multiple Routing)

Chapter 8. Example of Advanced Multiple Routing in a Mortgage Loan Process

This example demonstrates the advanced multiple route processing that occurs in a Mortgage Loan Department. It is modified from the example in the Work Management Builder tutorial. The example shows the processing that occurs in a mortgage loan department. The example is an artificial work process, created only to explain the concepts surrounding work process analysis and building.

Determining What Workbaskets Are Needed

Based on the information from the individual employee logs, identify the information needed to define:

- Work package contents
- Workbaskets
- Workbasket profiles
- Action lists

Determining Contents of a Work Package

In the daily logs, you requested that your employees list the documents and the document types related to the process. This list is necessary to determine the types of documents needed to complete the activities in your work process. In the mortgage loan example, a prospective home buyer submits the initial loan application. A credit and appraisal report are the supporting documents needed to complete the final work package. For the mortgage loan example, the following types of documents are required:

- A loan application
- A credit report
- An appraisal report

The tasks at each step in the work process are performed using only these three types of documents in the work package. The contents of the work package changes depending on the location of the work package in the mortgage loan process. See "Identifying the Flow of Work Across the Process" on page 75.

Determining Where the Documents Originate

In the daily logs, you requested that your employees list the origination point of the documents. See Table 8 on page 69. In the mortgage loan example, the initial loan application is the document that starts the work process. It is submitted by the prospective home buyer and scanned into the system by your employees.

The credit report and the appraisal report are added later to the work package that contains the initial loan application.

<i>Table 8. Document Origination Points</i>	
Document	Origination point
A loan application	Scanned into the system
A credit report	Scanned into the system
An appraisal report	Scanned into the system

Determining the Roles Each Employee Performs

In the daily logs, you requested that your employees list the role that they performed. You identified the following role responsibilities:

- Initiate new loan applications
- Review and process credit reports
- Review and process appraisal report
- Review late individual reports (exceptions)
- Evaluate entire loan application package (final review of the package)
- Review late group reports (exceptions)
- Send approval letter
- Send rejection letter

In the mortgage loan process, these responsibilities are assigned to individual workbaskets. For clarity, each workbasket in this example is associated with a single role. However, this is not the most efficient use of the work management function built into Workfolder Application Facility. After you become more proficient at analyzing your work process, combine workbaskets and action lists where appropriate. To facilitate the first steps of understanding a process analysis, this chapter focuses on the single mapping, and then identifying areas where combining workbaskets and actions lists is possible and beneficial.

In general, a workbasket is similar to a physical in-basket on an employee's desk that has a stack of documents or work in it. It is also similar to a Workfolder Application Facility case queue. However, in Workfolder Application Facility, the workbasket represents not only the stack of work in the workbasket, but also the actions that an employee performs on the work in the workbasket (using an action list). It also represents a special set of rules associated with that workbasket (using a workbasket profile).

For each workbasket, specify the following:

- The name of the workbasket
- The type of workbasket
- The rules for the workbasket
- The type of actions that employees perform

Name the Workbasket

Using the information that you collected from employee logs, identify the unique roles that your employees perform. Use the role titles as the names of the workbaskets. Workbasket names are limited to 10 characters, so you might have to abbreviate the

role title to 10 characters. Make a list of the workbasket names used in your process, as shown in Table 9 on page 70.

<i>Table 9. Roles and Workbasket Names</i>	
Employee Role	Workbasket Name
Scan and index mortgage applications	SCANINDX
Initiate new loan applications	INIT
Review and process credit reports	PROCCREDIT
Review and process appraisal reports	PROCAPP
Review late individual reports	EXCEPT
Evaluate entire loan application package	FINALRVW
Review late group reports	EXCEPT15
Send approval letters	APPRVLTR
Send rejection letters	REJECTLTR

Selecting the Correct Type of Workbasket

Based on the type of work done (the employees' role) at the workbasket, choose one of the six different predefined types of workbaskets, or define a workbasket yourself. For most of the workbaskets, the Work with work packages Type 6 workbasket is used.

Table 10 contains the employee roles, workbasket names, and workbasket types.

<i>Table 10 (Page 1 of 2). Roles, Workbasket Names, and Workbasket Types</i>		
Employee Role	Workbasket Name	Workbasket Type
Scan and index mortgage applications	SCANINDX	Type 4 - Scan and index into folders
Initiate new loan applications	INIT	Type 06 - Work with work packages
Review credit reports	PROCCREDIT	Type 06 - Work with work packages
Review appraisal report	PROCAPP	Type 06 - Work with work packages
Review late individual reports	EXCEPT	Type 06 - Work with work packages
Evaluate entire loan application package	FINALRVW	Type 06 - Work with work packages
Review late group reports	EXCEPT15	Type 06 - Work with work packages
Send approval letters	APPRVLTR	Type 06 - Work with work packages

<i>Table 10 (Page 2 of 2). Roles, Workbasket Names, and Workbasket Types</i>		
Employee Role	Workbasket Name	Workbasket Type
Send rejection letters	REJECTLTR	Type 06 - Work with work packages

Note that the first workbasket SCANINDEX has the same name as that used in the first medical claims process. In fact, with Workfolder Application Facility you can use the same workbasket profile and action list in many different processes. You just have to specify the process for which the workbasket is intended.

Each type of workbasket has a different set of permissible actions and rules associated with it. The rules associated with the different types of workbaskets are defined in the next section.

Determining Security Restrictions and Tracing Activities

The rules or workbasket profiles let you specify many attributes of the workbasket. Dependent on the type of workbasket, the rules you can specify might include:

- The security level of the workbasket
- Whether information about employee actions is logged or not
- Whether information about accessing a workbasket is logged or not
- The report key fields that allow you to access the logged information using the AS/400 query function
- Whether the work packages are presented in a list to be selected or not
- Whether the documents are scanned and then indexed or whether the documents are indexed and then scanned

Determining Security Restrictions Surrounding Each Work Package

In the analysis of your daily logs, employees listed any security restrictions surrounding the documents and work package with which they work. In the mortgage loan example, you have three documents:

- A loan application
- A credit report
- An appraisal report

However, you cannot just consider a base security level as sufficient because at different stages in handling the work package, the security of the work package might change. In the mortgage loan example, during the initial processing of all documents, a base level of 100 is applied to the work package. However, once the initial review of the documents is complete, and a work package with all of the documents is created, the security level is increased to 150. Finally, when the loan is approved, a much higher level of security is placed on the work package to prevent non-authorized employees from accessing and approving loans. The security level at the approval stage is 300.

This means that you should have different security levels at each of the workbaskets you create. The security level prevents any employee with a lesser security level from accessing that workbasket or its contents. The security level of a workbasket is one of its rules that is specified using the workbasket profile. For information about document level security, refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: System Administration Guide*.

Table 11. Roles, Workbasket Names, and Workbasket Types, Security Level

Employee Role	Workbasket Name	Workbasket Type	Security Level
Scan and index mortgage applications	SCANINDEX	Type 4 - Scan and index into folders	0
Initiate new loan applications	INIT	Type 06 - Work with work packages	100
Review credit reports	PROCCREDIT	Type 06 - Work with work packages	100
Review appraisal report	PROCAPP	Type 06 - Work with work packages	100
Review late individual reports	EXCEPT	Type 06 - Work with work packages	100
Evaluate entire loan application package	FINALRVW	Type 06 - Work with work packages	150
Review late group reports	EXCEPT15	Type 06 - Work with work packages	150
Send approval letters	APPRVLTR	Type 06 - Work with work packages	300
Send rejection letters	REJECTLTR	Type 06 - Work with work packages	150

Determining the Trace Requirements for Each Workbasket

In the analysis of your daily logs, employees listed any activities that you have to track in a special log, history, or trace file. In the mortgage loan example, you have legal and security requirements to track all activities on the loan and each time an employee accesses a workbasket containing loan material.

In the analysis of the mortgage loan group logs, you listed the changes that occurred to the work package priority and identified the special handling that occurs throughout the process. Table 12 shows the logging and report key fields needed for each of the workbaskets.

Table 12 (Page 1 of 2). Workbasket Names, Logging, and Report Key Fields

Workbasket Name	Logging	Report Key Fields
SCANINDEX	no logging	N/A

Table 12 (Page 2 of 2). Workbasket Names, Logging, and Report Key Fields

Workbasket Name	Logging	Report Key Fields
INIT	<ul style="list-style-type: none"> History - work package access User activity - workbasket access 	<ul style="list-style-type: none"> MORTGAGE CREDIT APPRAISAL
PROCCREDIT	<ul style="list-style-type: none"> History - work package access User activity - workbasket access 	<ul style="list-style-type: none"> MORTGAGE CREDIT APPRAISAL
PROCAPP	<ul style="list-style-type: none"> History - work package access User activity - workbasket access 	<ul style="list-style-type: none"> MORTGAGE CREDIT APPRAISAL
EXCEPT	no logging	N/A
FINALRVW	no logging	N/A
EXCEPT15	no logging	N/A
APPRVLTR	no logging	N/A
REJECTLTR	no logging	N/A

Determining the Actions Associated with Each Task Employees Perform

In the analysis of your daily logs, employees listed all of the specific actions that occur to each document. In the mortgage loan example, the following major actions are associated with each work package:

Initial loan application	Modify, save, retrieve, reject
Credit report	Approve, reject
Appraisal report	Approve, reject
Review and evaluation	Approve, reject
Approval letter	Create, send letter, file
Rejection letter	Create, send letter, file

In addition to the main task that employees perform, they must be permitted to perform other, necessary, actions. Table 13 contains the names of the workbaskets, the names of the actions lists, the action list types, and a list of actions associated with each workbasket for the mortgage loan example.

Table 13 (Page 1 of 3). Workbasket Names, Action List Names, Action List Types, and Actions

Workbasket Name	Action List Name	Action List Type	Actions
SCANINDEX	SCANACTS	Index	<ul style="list-style-type: none"> Continue Exit

<i>Table 13 (Page 2 of 3). Workbasket Names, Action List Names, Action List Types, and Actions</i>			
Workbasket Name	Action List Name	Action List Type	Actions
INIT	EXCEPT2	Work	<ul style="list-style-type: none"> • Move document • Display document • Continue • Exit • Reassign • Cancel
PROCCREDIT	WORKAL	Work	<ul style="list-style-type: none"> • Move document • Copy document • Exit • Reassign
PROCAPP	PROCAPP	Work	<ul style="list-style-type: none"> • Move document • Copy document • Continue • Exit • Reassign
EXCEPT	EXCEPT15	Work	<ul style="list-style-type: none"> • Move document • Display document • Continue • Exit • Reassign • Cancel
FINALRVW	FINALRVW	Work	<ul style="list-style-type: none"> • Move document • Display document • Continue • Exit • Reassign • Cancel
EXCEPT15	EXCEPT15	Work	<ul style="list-style-type: none"> • Move document • Display document • Continue • Exit • Reassign • Cancel
APPRVLTR	APPRV_RJT	Work	<ul style="list-style-type: none"> • Move document • Display document • Continue • Exit • Customer-defined key • Reassign • Cancel

<i>Table 13 (Page 3 of 3). Workbasket Names, Action List Names, Action List Types, and Actions</i>			
Workbasket Name	Action List Name	Action List Type	Actions
REJECTLTR	APPRV_RJT	Work	<ul style="list-style-type: none"> • Move document • Display document • Continue • Exit • Customer-defined key • Reassign • Cancel

As shown in Table 13 on page 73, some of the actions on the action lists are identical. For instance, the actions for creating the approval and rejection letter are identical, only the end effect (output written by you) is different. This is a clear example of where a single action list can be created and used to represent the actions in multiple workbaskets (APPRVLTR and REJECTLTR). You can use the same action list in as many workbaskets as you want. There is no need to create separate action lists for each workbasket, unless the actions required are different.

Also in Table 13 on page 73 you can see that some lists of actions include customer-defined actions—see the REJECTLTR workbasket, for example. These special keys and options let you pass information about the work package to a user exit. For the approval letter and the rejection letter, you pass information to the letter generation program about the work package including user and document information.

Identifying the Flow of Work Across the Process

Based on your analysis of the logs from all of the employees, map out any changes that occur in the information that your department handles and the flow of work.

Determining the Final Product of the Work Process in Your Department

In the analysis of your group logs, you listed the main work product of your department. In the mortgage loan example, the final product is the letter sent to a client approving or rejecting a mortgage loan application. When you create the work process that represents the flow of work in the mortgage loan department, you are describing the activities that result in the final product.

Based on the example, label the process using up to 10 characters. This example is called MORTGAGE.

Determining the Path a Work Package Takes from Start to Finish

In the analysis of your group logs, you listed the paths or routes that each document follows. Based on the analysis, you connect those employees in your department who initiate a document with those employees who subsequently work on the work package.

In the mortgage loan example, you identified the employees who originated each document in the work package with the employees who are responsible for other activities after the origination points.

You create connections between your representation of the originating employees and the receiving employees.

Table 14. Originating and Receiving Employees

Originating	Receiving
Initiate new loan application	<ul style="list-style-type: none"> • Review credit report • Review appraisal
Review credit report	Review and evaluate package
Review appraisal	Review and evaluate package
Review and evaluate package	<ul style="list-style-type: none"> • Notice of approval • Notice of rejection

Determining Where in Your Work Process a Work Package Must Wait

In the analysis of your daily logs, employees listed if they had to wait for other documents or other information before continuing to work with that document. In addition, employees listed anytime in the process when one employee must or can handle a document at the same time as another employee and how those documents later become coordinated.

You create collection points to represent the places in your work process where a work package must wait or be held until specified documents arrive or specified events occur. For more details on collection points, see Chapter 4, “Understanding Collection Points” on page 20. The mortgage loan example analysis showed the employees who are responsible for the following:

- Reviewing the credit report. They receive a new work package when the loan application is sent from the INIT workbasket and the credit report is online. When both arrive, the work package containing the loan application and the credit report is sent to the PROCCREDIT workbasket. See Table 14. A collection point called WAITRPT coordinates this waiting.
- Reviewing the appraisal reports. They receive a new work package when the loan application is sent from the INIT workbasket and the appraisal report is online. When both arrive, the work package containing the loan application and the appraisal report is sent to the PROCAPP workbasket. See Table 14. A collection point called WAITAPPL coordinates this waiting.
- Reviewing the process of the entire loan package. They receive a new work package when the credit report people working the PROCCREDIT workbasket have finished their work on the work package and when the appraisal report people working the PROCAPP workbasket have completed their work. See Table 14. Note that this is the coordination of activity from two different workbaskets. The work package does not progress onto the final review until the employees at both

PROCCREDIT and PROCAPP have completed their work with the work package. A collection point called WAITDOCS coordinates this waiting.

Determining Maximum Time for Work Process Step

In the previous analysis step of the daily logs, you looked at whether the employees must wait for other documents or must wait for other employees to finish their work on the work packages. But what happens if the event does not occur? Does a work package wait forever or is it lost in limbo?

Crucial to waiting either for other documents to be added to the work package or for other steps in the process to be completed, is the maximum amount of time that a work package can be placed on hold (pending) before someone receives that work package. If a work package is waiting, there must be criteria for it to be released. There are two major methods for releasing a work package based on the occurrence of an event. The first major type of event was discussed in the previous step. See “Determining Where in Your Work Process a Work Package Must Wait” on page 76. The other type of event is the amount of time that passes.

With time criteria, you set up an alternative (exception) path where the work package goes if the time criteria is exceeded so that the work package does not wait forever and the work package is sent to someone to be handled properly.

In the mortgage loan example, you set up two different time criteria:

- For the WAITRPT and the WAITAPPL collection points, you determined that you will wait two days before sending the work package on to a special group of employees who handle missing information (the exceptions). The workbasket that you have assigned for this function is called EXCEPT because the employees working on the work packages handle the exceptions.
- For the WAITDOCS collection point, you will wait 15 days before sending the work package on to another group of employees who handle the incomplete work package. The workbasket that you have assigned for this function is called EXCEPT15 because the employees working on the work packages in it handle the exceptions.

Determining Any Special Handling or Routing Decisions

In the mortgage loan example the employees who perform the final review decide whether the loan should be approved or rejected. If the loan is approved, the work package is sent on to an approval team. If the loan is rejected, the work package is sent on to a reject team. For this action, you can set up the action list for the final review team to have function keys that they press that automatically route the work package to the appropriate workbasket. All that you have to specify is a continue function to allow the work package to either continue on to the approval team if the function key is F6 or continue on to the reject team.

Determining Whether the Action Is Manual or Based on a Program

All actions in this work process are manual, performed from workbaskets.

Determining Where Work Package Contents Go When the Work Process Is Complete

In the analysis of your group logs, you listed what happens to a work package and the documents in it after your department is finished working on it. In the mortgage loan example, after your employees send a letter of loan approval or rejection, the work package containing the loan application, credit report, appraisal, and final decision is filed. You use the STOP command to signify that you are finished with the process, and that no further action is necessary. The work package is filed in the person's folder.

The following sections present the formal definition of the information that has been identified in the analysis of the Mortgage Loan Process. The specifics of each object are presented, including the following:

- Workbaskets and their associated profiles
- Action lists and their types
- Other objects needed to define the work flow
- The flow of information

Defining the Mortgage Loan Process

With the information outlined in the previous sections, it is now possible to completely define the mortgage loan process.

Workbaskets and Their Profiles

In the mortgage loan example, you defined nine workbaskets. The workbaskets you defined are as follows:

- SCANINDEX
- INIT
- PROCAPP
- PROCCREDIT
- EXCEPT
- FINALRVW
- EXCEPT15
- REJECTLTR
- APPRVLTR

The details of the definition of each workbasket are shown in the following figures. (See Figure 14 on page 45 for the information about the SCANINDEX workbasket.)

INIT Workbasket Profile

Figure 30 on page 79 shows the details needed to define the INIT workbasket. Note the following items:

- The security level is set to 100. If a user's security range does not include this level of security, the user is not able to access this workbasket. A user's workbasket security range is defined in the user profile.
- Both the history and the user activity is logged. The log information can be accessed using the three Report keys (MORTGAGE, CREDIT, and APPRAISAL). You use the AS/400 Query to access information recorded in the logs.
- The workbasket type is a general Type 6, Work with work packages.
- The action list is called EXCEPT2. It is defined as a Type 2 - Work action list.
- The work packages are shown in a list so that users can select which one they want to work with.
- The work packages are worked in first in, first out (FIFO) order. The work packages that arrive first are worked first.
- Because this is not a scan and index type workbasket, scan before indexing is not an option.
- Work can be reassigned to this workbasket from other workbaskets in the process.

Object type	: Workbasket
Name	: INIT
Description	: Initial wb, review info
Security class	: 100
History log level	: Work package access
User activity log level	: Work basket access
Report key	: MORTGAGE
	: CREDIT
	: APPRAISAL
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: EXCEPT2
List work package	: Yes
Work order	: First in first out
Scan before index	: No
Allow reassign to	: Yes

Figure 30. Workbasket Where Review of the Initial Loan Application Occurs

PROCAPP Workbasket Profile

Figure 31 on page 80 shows the details needed to define the PROCAPP workbasket. Note the following items:

- The security level is set to 100. If a user's security range does not include this level of security, the user is not able to access this workbasket.

- Both the history and the user activity is logged. The log information can be accessed using the three Report keys (MORTGAGE, CREDIT, and APPRAISAL). You use the AS/400 Query to access information recorded in the logs.
- The workbasket type is a general Type 6, Work with work packages.
- The action list is called PROCAPP. It is a Type 2 - Work action list.
- The work packages are shown in a list so that users can select which one they want to work with.
- The work packages are worked in first in, first out (FIFO) order. The work packages that arrive first are worked first.
- Since this is not a scan and index type workbasket, scan before indexing is not an option.
- Work can be reassigned to this workbasket from other workbaskets in the process.

Object type	: Workbasket
Name	: PROCAPP
Description	: Process appraisal
Security class	: 100
History log level	: Work package access
User activity log level	: Work basket access
Report key	: MORTGAGE
	: CREDIT
	: APPRAISAL
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: PROCAPP
List work package	: Yes
Work order	: First in first out
Scan before index	: No
Allow reassign to	: Yes

Figure 31. Workbasket Where Review of the Appraisal Occurs

PROCCREDIT Workbasket Profile

Figure 32 on page 81 shows the details needed to define the PROCCREDIT workbasket. Note the following items:

- The security level is set to 100. If a user's security range does not include this level of security, the user is not able to access this workbasket.
- Both the history and the user activity is logged. The log information can be accessed using the three Report keys (MORTGAGE, CREDIT, and APPRAISAL). You use the AS/400 Query to access information recorded in the logs.
- The workbasket type is a general Type 6, Work with work packages.

- The action list is called WORKAL. It is a Type 2 - Work action list.
- The work packages are shown in a list so that users can select which one they want to work with.
- The work packages are worked in first in, first out (FIFO) order. The work packages that arrive first are worked first.
- Since this is not a scan and index type workbasket, scan before indexing is not an option.
- Work can be reassigned to this workbasket from other workbaskets in the process.

Object type	: Workbasket
Name	: PROCCREDIT
Description	: Process credit report
Security class	: 100
History log level	: Work package access
User activity log level	: Work basket access
Report key	: MORTGAGE
	: CREDIT
	: APPRAISAL
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: WORKAL
List work package	: Yes
Work order	: First in first out
Scan before index	: No
Allow reassign to	: Yes

Figure 32. Workbasket Where Review of the Credit Report Occurs

EXCEPT Workbasket Profile

Figure 33 on page 82 shows the details needed to define the EXCEPT workbasket.

Note the following items:

- The security level is set to 100. If a user's security range does not include this level of security, the user is not able to access this workbasket.
- Neither the history nor the user activity log are logged.
- The workbasket type is a general Type 6, Work with work packages.
- The action list is called EXCEPT15. It is a Type 2 - Work action list.
- Work packages are not listed. Users work on the next available work package without being able to select from a list of work packages.
- The work packages are worked in first in, first out (FIFO) order. The work packages that arrive first are worked first.

- Since this is not a scan and index type workbasket, scan before indexing is not an option.
- Work can be reassigned to this workbasket from other workbaskets in the process.

Object type	: Workbasket
Name	: EXCEPT
Description	: Contact appraiser
Security class	: 100
History log level	: No logging
User activity log level	: No logging
Report key	:
	:
	:
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: EXCEPT15
List work package	: No
Work order	: First in first out
Scan before index	: No
Allow reassign to	: Yes

Figure 33. Workbasket for Exception Processing If the Appraisal and Credit Report Are Not Received within Time Allowed

FINALRVW Workbasket Profile

Figure 34 on page 83 shows the details needed to define the FINALRVW workbasket.

Note the following items:

- The security level is set to 150. If a user's security range does not include this level of security, the user is not able to access this workbasket.
- Neither the history nor the user activity log are logged.
- The workbasket type is a general Type 6, Work with work packages.
- The action list is called FINALRVW. It is a Type 2 - Work action list.
- Work packages are not listed. Users work on the next available work package without being able to select from a list of work packages.
- The work packages are worked in first in, first out (FIFO) order. The work packages that arrive first are worked first.
- Since this is not a scan and index type workbasket, scan before indexing is not an option.
- Work can be reassigned to this workbasket from other workbaskets in the process.

Object type	: Workbasket
Name	: FINALRVW
Description	: Final review
Security class	: 150
History log level	: No logging
User activity log level	: No logging
Report key	:
	:
	:
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: FINALRVW
List work package	: No
Work order	: First in first out
Scan before index	: No
Allow reassign to	: Yes

Figure 34. Workbasket Where Review of the Work Package Occurs for Approval or Rejection of the Mortgage Loan

EXCEPT15 Workbasket Profile

Figure 35 on page 84 shows the details needed to define the EXCEPT15 workbasket. Note the following items:

- The security level is set to 150. If a user's security range does not include this level of security, the user is not able to access this workbasket.
- Neither the history nor the user activity log are logged.
- The workbasket type is a general Type 6, Work with work packages.
- The action list is called EXCEPT15. It is a Type 2 - Work action list.
- Work packages are not listed. Users work on the next available work package without being able to select from a list of work packages.
- The work packages are worked in first in, first out (FIFO) order. The work packages that arrive first are worked first.
- Since this is not a scan and index type workbasket, scan before indexing is not an option.
- Work can be reassigned to this workbasket from other workbaskets in the process.

Object type	: Workbasket
Name	: EXCEPT15
Description	: 15 days late
Security class	: 150
History log level	: No logging
User activity log level	: No logging
Report key	:
	:
	:
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: EXCEPT15
List work package	: No
Work order	: First in first out
Scan before index	: No
Allow reassign to	: Yes

Figure 35. Workbasket for Exception Processing if the Work Package Is Not Received within 15 Days

REJECTLTR Workbasket Profile

Figure 36 on page 85 shows the details needed to define the REJECTLTR workbasket. Note the following items:

- The security level is set to 150. If a user's security range does not include this level of security, the user is not able to access this workbasket.
- Neither the history nor the user activity log are logged.
- The workbasket type is a general Type 6, Work with work packages.
- The action list is called APPRV_RJT. It is called APPRV_RJT because the same action list is used in both the approval and reject letter workbaskets. It is a Type 2 - Work action list.
- Work packages are not listed. Users work on the next available work package without being able to select from a list of work packages.
- The work packages are worked in first in, first out (FIFO) order. The work packages that arrive first are worked first.
- Since this is not a scan and index type workbasket, scan before indexing is not an option.
- Work can be reassigned to this workbasket from other workbaskets in the process.

Object type	: Workbasket
Name	: REJECTLTR
Description	: Send reject letter
Security class	: 150
History log level	: No logging
User activity log level	: No logging
Report key	:
	:
	:
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: APPRV_RJT
List work package	: No
Work order	: First in first out
Scan before index	: No
Allow reassign to	: Yes

Figure 36. Workbasket Where the Letter to Reject the Loan Application Is Generated

APPRVLTR Workbasket Profile

Figure 37 on page 86 shows the details needed to define the APPRVLTR workbasket.

Note the following items:

- The security level is set to 300. If a user's security range does not include this level of security, the user is not able to access this workbasket.
- Neither the history nor the user activity log are logged.
- The workbasket type is a general Type 6, Work with work packages.
- The action list is called APPRV_RJT. It is a Type 2 - Work action list. It is the same action list that is used with the REJECTLTR workbasket.
- Work packages are not listed. Users work on the next available work package without being able to select from a list of work packages.
- The work packages are worked in first in, first out (FIFO) order. The work packages that arrive first are worked first.
- Since this is not a scan and index type workbasket, scan before indexing is not an option.
- Work can be reassigned to this workbasket from other workbaskets in the process.

Object type	: Workbasket
Name	: APPRVLTR
Description	: Send approval letter
Security class	: 300
History log level	: No logging
User activity log level	: No logging
Report key	:
	:
	:
Type	: Work with work packages
Work process	:
File cabinet	:
Document type	:
Action list	: APPRV_RJT
List work package	: No
Work order	: First in first out
Scan before index	: No
Allow reassign to	: Yes

Figure 37. Workbasket Where a Letter Is Generated to Approve the Loan Application

Action Lists and Their Types

In the mortgage loan example, you defined seven action lists. The action lists that you defined are as follows:

- SCANACTS
- APPRV_RJT
- EXCEPT2
- EXCEPT15
- FINALRVW
- PROCAPP
- WORKAL

The details of each action list are given in the following sections. For details of the SCANACTS action list, see Figure 16 on page 46.

APPRV_RJT Action List

Figure 38 on page 87 shows the details needed to define the APPRV_RJT action list. Note the following items:

- The action list is defined as a Type 2 - Work action list.
- The only actions that users are allowed to perform are as follows:
 - Move** To move the document to another case or file cabinet.
 - Display** To display the document on the image display.
 - Continue** To allow the user to leave the work package and have the work package sent along the next step in the defined route. You have labelled the key Send letter to signify that a letter is sent. The users

who perform this activity use another program to actually generate the letter. The program was not defined as a part of the process since the mortgage loan company writes each letter individually.

- Exit** To exit the work package and exit the function.
- Reassign** To send the work package to another workbasket not defined in the current path.
- Cancel** To leave the current work package, but not the function.

Object type	: Action List	
Name	: APPRV_RJT	
Description	: Write/send letter	
Type	: Work	
List of functions	:	
Description	Number	Prompt
-----	-----	-----
Move document	2	Move
Display document	5	Display
Exit key	F3	Exit
Continue key	F6	Send letter
Reassign key	F7	Reassign
Cancel key	F12	Cancel

Figure 38. Actions Allowed for the Workbaskets Where a Letter Is Generated to Approve or Reject the Mortgage Loan

EXCEPT2 Action List

Figure 39 on page 88 shows the details needed to define the EXCEPT2 action list. Note the following items:

- The action list is defined as a Type 2 - Work action list.
- The only actions that users are allowed to perform are as follows:
 - Move** To move the document to another case or file cabinet.
 - Display** To display the document on the image display.
 - Continue** To allow the user to leave the work package and have the work package sent along the next step in the defined route.
 - Exit** To exit the work package and exit the function.
 - Reassign** To send the work package to another workbasket not defined in the current path.
 - Cancel** To leave the current work package, but not the function.

Object type	: Action List	
Name	: EXCEPT2	
Description	: Contact appraiser for info	
Type	: Work	
List of functions	:	
	Number	Prompt
	-----	-----
Move document	2	Move
Display document	5	Display
Continue key	F2	Continue
Exit key	F3	Exit
Reassign key	F7	Reassign
Cancel key	F12	Cancel

Figure 39. Actions Allowed for the INIT Workbasket

EXCEPT15 Action List

Figure 40 on page 89 shows the details needed to define the EXCEPT15 action list. Note the following items:

- The action list is defined as a Type 2 - Work action list.
- The only actions that users are allowed to perform are as follows:
 - Move** To move the document to another case or file cabinet.
 - Display** To display the document on the image display.
 - Continue** To allow the user to leave the work package and have the work package sent along the next step in the defined route.
 - Exit** To exit the work package and exit the function.
 - Reassign** To send the work package to another workbasket not defined in the current path.
 - Cancel** To leave the current work package, but not the function.

Because it is identical to the EXCEPT2 action list, you could have used the EXCEPT2 action list instead of defining a new action list.

Object type	: Action List	
Name	: EXCEPT15	
Description	: Contact procapp	
Type	: Work	
List of functions	:	
Description	Number	Prompt
-----	-----	-----
Move document	2	Move
Display document	5	Display
Continue key	F2	Continue
Exit key	F3	Exit
Reassign key	F7	Reassign
Cancel key	F12	Cancel

Figure 40. Actions Allowed for the Workbasket where Exception Processing occurs when the Work Package Is Not Received within 15 Days

FINALRVW Action List

Figure 41 on page 90 shows the details needed to define the FINALRVW action list. Note the following items:

- The action list is defined as a Type 2 - Work action list.
- The only actions that users are allowed to perform are as follows:
 - Move** To move the document to another case or file cabinet.
 - Display** To display the document on the image display.
 - Continue** To allow the user to leave the work package and have the work package sent along the next step in the defined route.
 - Exit** To exit the work package and exit the function.
 - Reassign** To send the work package to another workbasket not defined in the current path.
 - Cancel** To leave the current work package, but not the function.

Object type	: Action List	
Name	: FINALRVW	
Description	: Final review	
Type	: Work	
List of functions	:	
	Number	Prompt
	-----	-----
Move document	2	Move
Display document	5	Display
Continue key	F6	Approve
Exit key	F3	Exit
Reassign key	F7	Reassign
Cancel key	F12	Cancel

Figure 41. Actions Allowed for the Workbasket Where Final Review of the Work Package Occurs for Approval or Rejection of the Application

PROCAPP Action List

Figure 42 on page 91 shows the details needed to define the PROCAPP action list. Note the following items:

- The action list is defined as a Type 2 - Work action list.
- The only actions that users are allowed to perform are as follows:
 - Move** To move the document to another case or file cabinet.
 - Copy** To copy the document to another case or file cabinet.
 - Continue** To allow the user to leave the work package and have the work package sent along the next step in the defined route.
 - Exit** To exit the work package and exit the function.
 - Reassign** To send the work package to another workbasket not defined in the current path.

Object type	: Action List	
Name	: PROCAPP	
Description	: Process appraisal	
Type	: Work	
List of functions	:	
Description	Number	Prompt
-----	-----	-----
Move document	2	Move
Copy document	3	Copy
Continue key	F2	Continue
Exit key	F3	Exit
Reassign key	F7	Reassign

Figure 42. Actions Allowed for the Workbasket Where the Appraisal is Processed

WORKAL Action List

Figure 43 on page 92 shows the details needed to define the WORKAL action list. Note the following items:

- The action list is defined as a Type 2 - Work action list.
- The only actions that users are allowed to perform are as follows:
 - Move** To move the document to another case or file cabinet.
 - Copy** To copy the document to another case or file cabinet.
 - Continue** To allow the user to leave the work package and have the work package sent along the next step in the defined route.
 - Exit** To exit the work package and exit the function.
 - Reassign** To send the work package to another workbasket not defined in the current path.

Because the WORKAL action list is identical to the PROCAPP action list, you could have eliminated either one and used it as an action list for both workbaskets.

Object type	: Action List	
Name	: WORKAL	
Description	: Process credit rpt	
Type	: Work	
List of functions	:	
Description	Number	Prompt
-----	-----	-----
Move document	2	Move
Copy document	3	Copy
Continue key	F2	Continue
Exit key	F3	Exit
Reassign key	F7	Reassign

Figure 43. Actions Allowed for the Workbasket Where the Credit Report Is Processed

Work Process Definition for Mortgage Loan Process

This section shows the flow of information based on both Work Management Builder and 5250 route commands.

Flow Based on Work Management Builder Symbols

Before you use Work Management Builder to create your work process, you should know that it is dependent on Workfolder Application Facility to download certain objects. Those objects are as follows:

- Document types
- File cabinets
- Workfolder Application Facility queues

After you have used the GUI to create the work process, you must do the following:

- **Verify that the process is correct.** The GUI has a list of rules that are used to verify that your work process works correctly.
- **Check in each object you created.** All of the objects that you created using Work Management Builder must be checked in to the AS/400 before they can be used. Do not forget to check in those objects not represented using symbols such as the scan and index workbaskets.

For more information about Work Management Builder, refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: User's Guide for the Work Management Builder*.

Figure 44 on page 93 shows the completed mortgage loan example process diagram built using Work Management Builder.

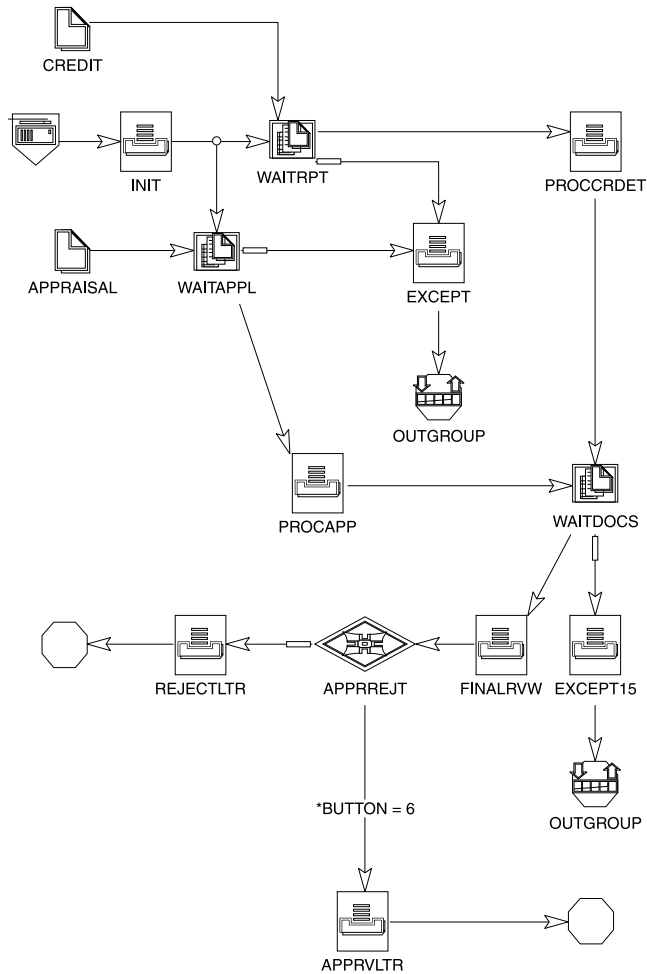


Figure 44. Completed Mortgage Loan Example

The symbols used in Work Management Builder to represent the mortgage loan example include the following:



Use a *begin node* to represent the beginning of the work process. It has no counterpart in Workfolder Application Facility.



Use a *work node* to represent a workbasket and associate it with an action list. Only workbaskets of Type 2, 3, 6, or 50-99 can be represented using a work node. The work process editor only displays Type 2, 3, 6, or 50-99 workbaskets in the Workbasket drop-down list of the Work Node dialog. The other workbasket types (1, 4, and 5) all involve scanning and are not represented using the work process editor because the work is actually performed by Workfolder Application Facility.

If you have defined either scan or index type workbaskets, they can be defined in the Workbasket and Action List portions of Work Management Builder. However, they are not represented by symbols. You must be sure that they are defined separately and then checked in to Workfolder Application Facility host to which you are connected.



Use a *decision point node* to represent where the paths in a work process change depending upon decisions coming into the decision point node.



Use a *collection point node* to represent where a work package waits for specific events. The events can be documents, previous workbaskets, or time.



Use a *document node* to represent a document in the work process.



Use a *stop node* to represent how the work process stops. A stop node indicates how the work process is stopped when it is run on Workfolder Application Facility. A stop node can represent a transfer of the work package contents to one of the following:

- The Workfolder Application Facility case processing queue
- A queue of closed cases
- The document library

For more information about Work Management Builder, refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: User's Guide for the Work Management Builder*.

Flow Based on Commands and Routes

Figure 45 shows the completed mortgage loan example process routes defined in Workfolder Application Facility. The completed example takes 12 separate routes to specify the information defined by the symbols shown in Figure 44 on page 93.

Object type	: Work Process
Name	: MORTGAGE
Description	: Mortgage loan example
Starting route name	: 10
Process item version	: 0
List of routes	:

Figure 45 (Part 1 of 3). Routes Used to Complete the Mortgage Loan Example

Route name				: 10
Step	Command	Parm 1	Parm 2	
10	WORK	INIT		
20	START	20		
30	START	100		
Route name				: 100
Step	Command	Parm 1	Parm 2	
10	COLLECT	WAITAPPL		
Route name				: 110
Step	Command	Parm 1	Parm 2	
10	WORK	PROCCAPP		
20	COLLECT	WAITDOCS		
Route name				: 20
Step	Command	Parm 1	Parm 2	
10	COLLECT	WAITRPT		
Route name				: 30
Step	Command	Parm 1	Parm 2	
10	WORK	PROCCREDIT	WORKAL	
20	COLLECT	WAITDOCS		
Route name				: 40
Step	Command	Parm 1	Parm 2	
10	WORK	EXCEPT15		
20	QUEUE	OUTGROUP		
Route name				: 50
Step	Command	Parm 1	Parm 2	
10	WORK	FINALRVW		
20	DECIDE	APPRREJT		

Figure 45 (Part 2 of 3). Routes Used to Complete the Mortgage Loan Example

Route name				: 60
Step	Command	Parm 1	Parm 2	
10	WORK	APPRVLTR		
20	STOP			
Route name				: 70
Step	Command	Parm 1	Parm 2	
10	WORK	REJECTLTR		
20	STOP			
Route name				: 90
Step	Command	Parm 1	Parm 2	
10	WORK	EXCEPT		
20	QUEUE	OUTGROUP		

Figure 45 (Part 3 of 3). Routes Used to Complete the Mortgage Loan Example

Collection Point Named WAITAPPL: The definition for WAITAPPL is shown in Figure 46 on page 97. It waits for the work package containing the loan application and the appraisal.

- Event 1 is the exception route. The work package follows this route if it has been waiting 2 days. After two days, the work package is sent to the Workfolder Application Facility case queue OUTGROUP along route 90.
- Event 2 waits on the arrival of a document type APPRAISAL and completion of route 100 from the INIT workbasket. When both occur, Event 2 sends the work package to the PROCAPP workbasket (route 110).

List of collection points	:
Collection point name	: WAITAPPL
Description	: Wait for appraisal
Event list ID	: 1
Completion route name	: 90
Exception condition	: Yes
Number of days	: 002
Event list ID	: 2
Completion route name	: 110
Wait on event	: Document type
Match criterion	: APPRAISL
Document event name	: APPRAISAL
Wait on event	: Work route
Match criterion	: 100

Figure 46. Collection Point Node that Waits for the Work Package Containing the Loan Application and the Appraisal

Collection Node Named WAITRPT: The definition for WAITRPT is shown in Figure 47 on page 98. It waits for the work package containing the loan application and the credit report.

- Event 1 is the exception route. The work package follows this route if it has been waiting 2 days. After two days, the work package is sent to the Workfolder Application Facility case queue OUTGROUP along route 90.
- Event 2 waits on the arrival of a document type of CREDIT and completion of route 20 representing processing from the INIT workbasket. When both occur, Event 2 sends the work package to the PROCCREDIT workbasket (route 30).

List of collection points	:
Collection point name	: WAITRPT
Description	: Wait for credit report
Event list ID	: 1
Completion route name	: 90
Exception condition	: Yes
Number of days	: 002
Event list ID	: 2
Completion route name	: 30
Wait on event	: Document type
Match criterion	: CREDIT
Document event name	: CREDIT
Wait on event	: Work route
Match criterion	: 20

Figure 47. Collection Point Node that Waits for the Work Package Containing the Loan Application and the Credit Report

Collection Point Named WAITDOCS: The definition for WAITDOCS is shown in Figure 48 on page 99. It waits for the work package from the two different workbasket reviews (PROCCREDIT and PROCAPP).

- Event 1 is the exception route. The work package follows this route if it has been waiting 15 days. After 15 days, the work package is sent to the Workfolder Application Facility case queue OUTGROUP along route 40.
- Event 2 waits on the completion of the two routes preceding it, routes 30 and 110, representing workbaskets PROCAPP and PROCCREDIT. When both occur, Event 2 sends the work package to the FINALRVW workbasket (route 50).

List of collection points	:
Collection point name	: WAITDOCS
Description	: Wait for processing
Event list ID	: 1
Completion route name	: 40
Exception condition	: Yes
Number of days	: 015
Event list ID	: 2
Completion route name	: 50
Wait on event	: Work route
Match criterion	: 30
Wait on event	: Work route
Match criterion	: 110

Figure 48. Collection Point Node that Waits for the Work Package Containing the Loan Appraisal and the Credit Report

Decision Point Named APPREJT: The definition for the decision point named APPREJT is shown in Figure 49. The decision has two branches.

The work package continues on from FINALRVW to the exception branch anytime the work package leaves FINALRVW without having the F6 key pressed. The work package continues on to the REJECTLTR workbasket (route 70).

The work package continues on from FINALRVW any time the employee who is working the FINALRVW workbasket presses the F6 key. The work package continues on to the APPRVLTR workbasket (route 60).

List of decision points	:
Decision point name	: APPREJT
Description	: Approve or reject the loan
Decision branch ID	: 1
Completion route name	: 70
Decision branch trigger	: Exception condition
Decision branch ID	: 2
Completion route name	: 60
Decision branch trigger	: *BUTTON EQ 0000000006

Figure 49. Decision Node that Represents the Decision of Whether the Loan Application is Sent for Approval or Rejection

Other Objects Needed to Define the Work Flow

The other objects that you identified as needed to define the mortgage loan example work flow are as follows:

- A document of type LOAN that represents the loan. It starts the process as it is scanned into the system.
- A document of type APPRAISAL that represents the loan appraisal report. It is already scanned into the system.
- A Workfolder Application Facility queue named OUTGROUP that represents the stop node where the work package is sent if exception processing occurs. The name OUTGROUP must represent an active queue.

For information about creating document types and using Workfolder Application Facility queues, refer to the *IBM ImagePlus Workfolder Application Facility for AS/400: System Administration Guide*.

Optimizing Workbaskets, Action Lists, and Work Processes

Thus far the analysis of a work process has been presented by identifying a single individual role with a single workbasket and a single action list. This is not the best way to get the most out of Workfolder Application Facility. The work management functions are designed to minimize redundancy.

For workbaskets, reexamine the roles that are defined and see if there are any that can be logically combined. For example, instead of separating the task of approve/reject then write letter, at the end of the process, consolidate the tasks into what happens at the FINALRVW workbasket. If the user at workbasket FINALRVW can make the decision whether to approve or reject, then they can probably also handle how to get the letter out to the customer. This would also eliminate the necessity of having a decision point (APPRVRJT).

In this example, a lot of redundancy was also created in the action lists. An action list can be used in many different workbaskets and work processes. It is best to create a single action list and only create another when necessary to enable the employee to perform different actions. The following action lists are identical and could be replaced with a single action list that performs the same actions:

- EXCEPT15, EXCEPT2, and FINALRVW
- PROCAPP and WORKAL

When you specify a work node or specify routes, you do not have to specify the action list associated with each workbasket because you already defined the relationship when you defined the workbasket. For work nodes in Work Management Builder there is an option called default. Specifying this means that the default action list, defined when the workbasket was defined, is used for every work package that comes into the workbasket. Similarly, when you define a route using the 5250 panels, you do not have to specify parameter 2 with an action list. You have already defined the action list associated with the workbasket when you defined the workbasket profile.

The only time that an action list must be specified separately is when you want to override the default action list (the action list that was defined when the workbasket was created).

Appendix A. Case Processing Functions and Work Management Equivalents

<i>Table 15 (Page 1 of 2). Case Processing Functions and Work Management Equivalents</i>		
Case Processing Function	Work Management Equivalent	How to Get to the Work Management Equivalent
Work any case	Search for work packages	<ol style="list-style-type: none"> 1. GO EKD200 2. Choose option 9 - Search for work packages 3. Type search criteria
Work queued case	Work with workbaskets	<ol style="list-style-type: none"> 1. GO EKD200 2. Choose option 7 - Work with workbaskets 3. Select workbasket Type 06 - Work with work packages
Review any case	Search for work packages	<ol style="list-style-type: none"> 1. GO EKD200 2. Choose option 9 - Search for work packages 3. Select workbasket Type 06 - Work with work packages
Scan documents only	Use a Scan documents type workbasket	<ol style="list-style-type: none"> 1. GO EKD100 2. Choose option 8 - Work with workbaskets 3. Select workbasket Type 01 - Scan documents
Scan documents into a single case	Use a Scan and index into folders type workbasket	<ol style="list-style-type: none"> 1. GO EKD100 2. Choose option 8 - Work with workbaskets 3. Select workbasket Type 04 - Scan and index into folders 4. In the workbasket chosen, set the Scan before index flag to N
Scan/capture and index	Use a Scan and index into folders type workbasket or use a Scan and index into file cabinets type workbasket	<ol style="list-style-type: none"> 1. GO EKD100 2. Choose option 8 - Work with workbaskets 3. Select workbasket Type 04 - Scan and index into folders or Type 05 - Scan and index into file cabinets
Index documents	Use an Index into folders type workbasket or use an Index into file cabinets type workbasket	<ol style="list-style-type: none"> 1. GO EKD100 2. Choose option 8 - Work with workbaskets 3. Select workbasket Type 02 - Index into folders or Type 03 - (Index into file cabinets)
Pend for a number of days	Use a collection point and use the Maximum wait days option	Define the collection point with the Maximum wait days only and point to the collection point with the COLLECT command in the appropriate route. Use pend function in action list to allow runtime definition.
Pend for a date	Pend key or use a collection point and use the number of days option	Function code 0115 in the action list or define the collection point with the Maximum wait days only and point to the collection point with the COLLECT command in the appropriate route. Use pend function in action list to allow runtime definition.

Table 15 (Page 2 of 2). Case Processing Functions and Work Management Equivalents

Case Processing Function	Work Management Equivalent	How to Get to the Work Management Equivalent
Pend for a document type	Pend key or use a collection point and use a list with an event type of 01 (document type) and the match criterion of the actual document type to wait for (or *ANY). Note: Match criterion is case-sensitive!	Function code 0115 in the action list or define the collection point with the appropriate event lists and point to the collection point with the COLLECT command in the appropriate route. Use pend function in action list to allow runtime definition.
Reindex	Reindex options	Function code 0007 or 0017 in the action list or you could use a RMVOBJ API through a customer-defined option.
Optical Only	Not directly supported in work management	You could use a customer-defined option and: <ul style="list-style-type: none"> • If the document is on DASD, delete it • If the document is not on optical, create a store request.
DASD and optical (option)	Retrieve document (option)	Function code 0012 in the action list
Close case	Close key	Function code 0216 in the action list
Release Pend	Release key or force continue from EKDWM213	Function code 0114 in the action list or: <ol style="list-style-type: none"> 1. GO EKD200 2. Choose option 9 - Search for work packages 3. Type the search criteria 4. When you find the work package instance in the collection point that you want to release, use force continue
Pend (dynamically here versus by the predefined work process)	Pend key	Function code 0115 in the action list
Add form to case	Add form to folder	

Bibliography

The following is a bibliography of related IBM publications that you might find helpful while using this book. See "Where to Find More Information" on page v for information about the VisualInfo for AS/400 product library.

You can request copies of IBM publications from your IBM representative or the IBM branch office serving your area.

IBM AS/400 Publications

Communications and Connectivity

- *IBM AS/400 Communications: Advanced Peer-to-Peer Configuration Guide*, GG24-4023
- *IBM AS/400 Communications: Advanced Peer-to-Peer Network User's Guide*, SC41-8188
- *IBM AS/400 Communications: Advanced Program-to-Program Communication Programmer's Guide*, SC41-8189
- *IBM AS/400 Communications: User's Guide and Reference*, SC09-1168
- *IBM AS/400 Communications Configuration*, SC41-3401
- *IBM AS/400 Communications Management*, SC41-3406
- *IBM AS/400 Network and Systems Management*, SC41-3409
- *IBM AS/400 Network Planning Guide*, GC41-9861

Languages

- *AS/400 Languages: COBOL/400 User's Guide*, SC09-1812
- *AS/400 Languages: COBOL/400 Reference*, SC09-1813
- *AS/400 Languages: COBOL/400 Reference Summary*, SX09-1285
- *AS/400 Languages: RPG/400 User's Guide*, SC09-1816
- *AS/400 Languages: RPG/400 Reference*, SC09-1817
- *ILE RPG/400 Reference Summary*, SX09-1261

- *ILE COBOL/400 Programmer's Guide*, SC09-1522
- *ILE COBOL/400 Reference*, SC09-1523
- *ILE COBOL/400 Reference Summary*, SX09-1260
- *AS/400 Programming: Control Language Programming*, SC41-3721
- *AS/400 Programming: Control Language Reference*, SC41-3722
- *ILE RPG/400 Programmer's Guide*, SC09-1525
- *ILE RPG/400 Reference*, SC09-1526
- *ILE RPG/400 Reference Summary*, SX09-1261

Planning, Installation, and Migration

- *IBM AS/400 Local Device Configuration*, SC41-3121
- *IBM AS/400: Physical Planning Guide*, GA41-9571
- *IBM AS/400 Software Installation*, SC41-3120
- *IBM AS/400 System Support: Installation Guide - 9404*, SY31-9066
- *IBM System Support AS/400: Installation Guide - 9406* (shipped with the product)

Programming

- *IBM AS/400 Programming: Backup and Recovery Guide*, SC41-8079
- *IBM AS/400 Programming*, SC41-3721
- *IBM AS/400 Programming: Security Concepts and Planning*, SC41-8083
- *Query/400: User's Guide*, SC41-3210

System Use

- *IBM AS/400 Getting Started on the AS/400 System*, SC41-3204
- *IBM AS/400 PC Support: Technical Reference*, SC41-8091
- *IBM AS/400 Q & A Database Coordinator's Guide*, SC41-8086
- *IBM AS/400 System Operation*, SC41-3203
- *IBM AS/400 System Operation for New Users*, SC41-3200
- *IBM AS/400 System Operations: Operator's Guide*, SC41-8082

- *IBM AS/400 System Startup and Problem Handling*, SC41-3206
- *OS/400 Infoseeker Getting Started*, SC41-3001
- *OS/400 Integrated File System Introduction*, SC41-3711

System Management

- *IBM AS/400 Security—Basic*, SC41-3301
- *IBM AS/400 Security—Reference*, SC41-3302
- *OS/400 Backup and Recovery—Basic*, SC41-3304
- *OS/400 Backup and Recovery—Advanced*, SC41-3305

IBM Client Access

Windows

- *IBM Client Access/400 for Windows: Getting Started*, SC41-3530
- *IBM Client Access/400 for Windows: User's Guide*, SC41-3532

OS/2

- *Client Access/400 for OS/2 User's Guide*, SC41-3521
- *Client Access/400 for OS/2 Setup*, SC41-3520

IBM Data Stream and Object Architectures

- *Data Stream and Object Architectures: Image Object Content Architecture Reference*, SC31-6805
- *Data Stream and Object Architectures: Mixed Object Document Content Architecture Reference*, SC31-6802
- *Data Stream and Object Architectures: Presentation Text Object Content Architecture Reference*, SC31-6803

IBM Facsimile Support for OS/400

- *IBM Facsimile Support for OS/400: Installation Guide*, SC41-0570
- *IBM Facsimile Support for OS/400: Programmer's Guide and Reference*, SC41-0572

- *IBM Facsimile Support for OS/400: User's Guide and Reference*, SC41-0571

IBM ImagePlus VisualInfo

- *IBM ImagePlus VisualInfo Getting Started*, GC31-9051
- *IBM ImagePlus VisualInfo Planning and Installation Guide*, GC31-7772
- *IBM ImagePlus VisualInfo System Administration Guide*, GC31-7774
- *IBM ImagePlus VisualInfo Application Programming Guide for Windows*, GC31-9055
- *IBM ImagePlus VisualInfo Application Programming Guide, Volume 1*, GC31-9063
- *IBM ImagePlus VisualInfo Application Programming Guide, Volume 2*, GC31-9061

IBM ImagePlus Workstation Program

Windows NT

- *IBM ImagePlus Workstation Program for Windows NT: Installation and User's Guide*, GC34-3102

OS/2

- *ImagePlus Workstation Program for OS/2: Programmer's Guide*, SC34-3093
- *IBM ImagePlus Workstation Program for OS/2: Installation and User's Guide*, GC34-3092

DOS for Windows

- *IBM ImagePlus Workstation Program/DOS for Windows Installation Guide*, SC09-1550
- *IBM ImagePlus Workstation Program/DOS for Windows User's Guide*, SC09-1549

IBM ImagePlus Intelligent Forms Facility

- *IBM ImagePlus Intelligent Forms Facility/2 Administrator's Guide for the ImagePlus/400 Environment*, SB35-0824
- *IBM ImagePlus Intelligent Forms Facility/2 Quick Reference for the ImagePlus/400 Environment*, SX76-0149

IBM ImagePlus Intelligent Forms Assist

- *IBM ImagePlus Intelligent Forms Assist/2 Installation and Configuration Guide for the ImagePlus/400 Environment* SB35-0831
- *IBM ImagePlus Intelligent Forms Assist/2 User's Guide for the ImagePlus/400 Environment*, SB35-0834

IBM Personal Communications

- *IBM Personal Communications Version 4.1 for OS/2 Up and Running*, SC31-8258
- *IBM Personal Communications Version 4.1 Tell Me About OS/2*, SC31-8257
- *IBM Personal Communications Version 4.1 for OS/2 Reference*, SC31-8259
- *IBM Personal Communications Version 4.1 for OS/2 Programmer's Guide*, SC31-8260

IBM 3995 Compact Optical Library Dataserver

Direct-Attached

- *IBM AS/400 Optical Library Dataserver Support/400: User's Guide and Reference*, SC41-0035
- *OS/400 Optical Support*, SC41-4310

LAN-Attached

- *IBM 130 mm Rewritable Optical Disk Cartridge Requirements*, SA37-0261
- *IBM 3995 All LAN Models Reference*, GA32-0147
- *IBM 3995 AS/400 Optical Library Dataserver: Operator's Guide Models A43, 143, 142, 043, and 042*, GA32-0140
- *IBM 3995 Optical Library Dataserver Products: Introduction and Planning Guide for C-Series Models*, GA32-0350
- *IBM 3995 Optical Library Dataserver Products: Model A23 Guide to Operations*, GA32-0291
- *IBM 3995 Optical Library Dataserver Products: Operator's Guide for C-Series Models*, GA32-0352
- *IBM 3995 Optical Library Dataserver Products: User's Guide for Models 123, 122, 121, 113, 112, 111, 023, 022, 021*, GA32-0141

Both Direct- and LAN-Attached

- *IBM 3995 Optical Library Dataserver Products: Introduction and Planning Guide*, GA32-0121
- *IBM 3995 Optical Library Dataserver Products: Optical Disk Format*, GA32-0224
- *IBM 3995 Optical Library Dataserver Products: Optical Disk Cartridge Requirements 130 mm Write-Once 1024 Bytes/Sector*, GA32-0146
- *IBM 3995 Optical Library Dataserver Products: Reference for C-Series Models*, GA32-0351
- *IBM 3995 Optical Library Dataserver Products: Safety*, GA32-0148

Glossary

This glossary defines terms and abbreviations used in this book and the product document library. Refer to the *IBM Dictionary of Computing*, ZC20-1699-09, for terms or abbreviations that do not appear here.

The following cross-references are used in this glossary:

- **Contrast with.** This refers to a term that has an opposed or substantively different meaning.
- **See.** This refers the reader to multiple-word terms in which this term appears.
- **See also.** This refers the reader to terms that have a related, but not synonymous, meaning.
- **Synonym for.** This indicates that the term has the same meaning as a preferred term, which is defined in the glossary.

A

access list. A list consisting of one or more individual user IDs or user groups and the *privilege set* associated with each user ID or user group. You use access lists to control user access to items in VisualInfo for AS/400. The items that can be associated with access lists are the data objects stored by users, *index classes* and subsets, *workbaskets*, and *workflows*.

action list. In Workfolder Application Facility work management, an approved list of the actions, defined by a supervisor, that a user can perform on work packages. An action list defines options, such as printing or displaying work packages, and the function keys that are available for use.

active case. A case that was entered and indexed, assigned to a queue, and has been previously pended, unqueued, assigned to a process, or assigned to a workbasket. A case with this status can be accessed through the Work any case, Work queued case, or Review any case menu options.

address ID profile. A control file used in Workfolder Application Facility that contains names and addresses.

ad hoc route. A route that is not part of a defined Workfolder Application Facility work management process. An *ad hoc route* is started when a user creates

a work package and assigns it directly to a workbasket. The user manually routes the work package from one workbasket to another by reassigning it.

administrator. The person responsible for system management, controls, and security, as well as case statistics. Synonymous with system administrator.

Advanced Peer-to-Peer Networking (APPN). Data communications support that routes data in a network between two or more APPC systems that are not directly attached.

advanced program-to-program communications (APPC). Data communications support that allows programs on an AS/400 system to communicate with programs on other systems having compatible communications support. This communications support is the AS/400 method of using the SNA LU session type 6.2 protocol.

annotation. An added descriptive comment or explanatory note.

APAR. Authorized Program Analysis Report.

API. Application programming interface.

application programmer. A programmer who designs programming systems and other applications for a user's system.

application program interface (API). The formally-defined programming language interface which is between an IBM system control program or a licensed program and the user of the program.

APPC. Advanced program-to-program communications.

APPN. Advanced Peer-to-Peer Networking.

application program interface (API). The formally-defined programming language interface which is between an IBM system control program or a licensed program and the user of the program.

archiving. The storage of backup files and any associated journals, usually for a given period of time.

AS/400. Application System/400.

AS/400 object directory profile. A control file used in Workfolder Application Facility to identify AS/400 object directories used for image document storage.

attribute. Used in VisualInfo for AS/400 APIs, a single value associated with an item (document or folder). Each index class can have up to eight attributes.

automatic importing. The process that operates in the background to complete the importing of documents when it is requested through the facsimile process or the "set up automatic import only" option.

automatic indexing. The indexing process that operates in the background to complete the indexing of documents introduced to the system through the scan and batch index function.

B

binary large object (BLOB). A large stream of binary data treated as a single object.

C

capture. In optical character recognition, to gather picture data from a field on an input document, using a special scan.

cartridge. (1) A storage device that consists of magnetic tape, on supply and takeup reels, in a protective housing. (2) For optical storage, a plastic case that contains and protects optical disks, permitting insertion into an optical drive. See also *optical disk* and *cartridge storage slots*.

cartridge storage slots. An area in an optical library where cartridges are stored.

case. A uniquely identifiable work item initiated by a user. Cases may be: Active (open and being worked), Pended (suspended awaiting further information), Closed (processing is complete), New, or Not Queued.

case file. A file that contains one entry for each case.

case history file. A file that contains the history of actions against each case.

case ID. A system-assigned identifier that is chronological, based on the time that indexing occurred. Every case has a unique case ID.

client application. An application written with the VisualInfo for AS/400 APIs to customize a user interface.

closed case. A case that was closed and cannot be reopened. A closed case is not assigned to a queue. A case with this status can be accessed through the Review any case menu option.

closing a case. Ending a case permanently. Documents in a closed case may be archived for future access.

collection. (1) In VisualInfo for AS/400, a group of objects with a similar set of storage management rules and contained within a *storage group*. Every object is stored in a collection. (2) In Workfolder Application Facility, it provides categories for entered documents and is used to match incoming documents with outstanding requests. Primarily used in case processing, collections can also be used when indexing documents into file cabinets.

collection point. In Workfolder Application Facility work management, the point where work packages wait for specific events to either occur or become synchronized before processing can continue. A collection point is part of a work process. For example, a collection point is where work packages that are part of the "open a new account" work process must wait until credit information is verified. See also *decision point*.

collection profile. A file that contains one entry for each type of document to be processed.

content class. A number that indicates the data format of an object, such as MO:DCA, TIFF, or ASCII.

control files. Files that govern the categories of work performed by an operator and the types of documents the system recognizes.

convenience workstation. A display workstation equipped with a printer and a scanner.

cross-system importing. The process that imports cases and documents to one Workfolder Application Facility system from another.

cross-system exporting. The process that exports cases and documents from one Workfolder Application Facility system to another.

current document. A document that is being processed.

customization. The process of designing a data processing installation or network to meet the requirements of particular users.

D

DASD. Direct access storage device.

DASD system ID profile. A file used to define the roles of AS/400 processors in the Workfolder Application Facility system.

DDM. Distributed data management.

DBCS. Double-byte character set.

decision point. In Workfolder Application Facility work management, the point where work packages continue on their current route or switch to an alternate route, depending on the specific information in each work package. Decision points are tables consisting of variable names, values, and routes. A decision point is part of a work process. For example, a decision point is where work packages that are part of the "open a new account" work process receive approval or not based on credit information.

See also *collection point*.

destager. A function of the object server that moves objects from the *staging area* to the first step in the object's *migration policy* or *management class*.

direct access storage device (DASD). A device in which access time is effectively independent of the location of the data.

distributed data management (DDM). A feature of the System Support Program that lets an application program work on files that reside in a remote system.

display workstation. An image processing workstation used primarily for displaying documents that have been previously scanned or imported into the AS/400 system.

document. (1) An item containing one or more base parts. See also *collection point*. (2) Information and the medium on which it is recorded that generally have permanence and that humans or machines can read. (3) A named, structural unit of text that can be stored, retrieved, and exchanged among systems and users as a separate unit. Also referred to as an *object*. A single document can contain many different types of base parts, including text, images, and objects such as spreadsheet files.

document content architecture (DCA). An architecture that guarantees information integrity for a document being interchanged in an office system network. DCA provides the rule for specifying form and meaning of a document. It defines revisable form text (changeable) and final form text (unchangeable).

document type. Provides categories for entered documents and are used to match incoming documents with outstanding requests. Primarily used in case processing, document types can also be used when indexing documents into file cabinets.

document type profile. A file that contains one entry for each type of document to be processed.

document working set. A set of document images selected from a menu that Workfolder Application Facility provides. This set of document images is sent to the ImagePlus workstation for display.

double-byte character set (DBCS). A set of characters in which each character occupies two bytes. Languages, such as Japanese, Chinese, and Korean, that contain more symbols than can be represented by 256 code points, require double-byte character sets. Entering, displaying, and printing DBCS characters requires special hardware and software support.

E

empty case. A case that has no documents.

export. In Workfolder Application Facility, a process used to write data from a document in a system folder to a file. Export and import processes can be used to transfer documents among Workfolder Application Facility systems.

F

facsimile configuration profile. A control file that lets the system administrator associate Facsimile Support for OS/400 with Workfolder Application Facility.

fax exporting. A process that operates in the background to send documents requested for fax transmission to a fax server.

fax importing. The importing process that operates in the background to forward fax documents received by the fax server.

FIFO (first in, first out). A queueing technique in which the next item to be retrieved is the item that has been in the queue for the longest time.

file cabinet. A component of Workfolder Application Facility that provides document storage and retrieval capabilities designed to help manage selected documents.

file cabinet code. Acts as the file cabinet name.

first in first out (FIFO). A queueing technique in which the next item to be retrieved is the item that has been in the queue for the longest time.

focus control. In Workfolder Application Facility, a feature that lets the system administrator decide whether Workfolder Application Facility and ImagePlus Workstation Program or the user controls the active window.

folder. (1) In VisuallInfo for AS/400, an object that can contain other folders or documents. (2) In Workfolder Application Facility, the area in the AS/400 system where images are stored after successful scanning or importing.

folder balancing. In the AS/400, the process by which documents are distributed evenly among the available folders in the system.

folder manager. In VisuallInfo systems other than VisuallInfo for AS/400, the term used to describe the data model and a subset of the APIs. In VisuallInfo for AS/400, this term refers to the entire set of VisuallInfo for AS/400 APIs.

folder name. A 1- to 12-character user-defined word that names a folder. One period (.) is allowed. If the folder name is more than 8 characters, the ninth character must be a period. This can be followed by a 1- to 3-character extension.

folder path. A folder name, followed by one or more additional folder names, where each preceding folder is found. Each folder in the path must be separated by a slash (/). A folder path can consist of 1 to 63 characters.

forms creation utility. A utility that allows the system administrator to capture a preprinted form as an image and to define the content and format of the form.

G

Group III. A compression algorithm that conforms to a standard promulgated by the International Telegraph and Telephone Consultative Committee (CCITT).

H

high-speed indexing. The indexing process that operates in the background to complete the indexing of documents that were input into the system using the high-speed scanning function.

high-speed scanner workstation. A display workstation equipped with a high-speed scanner.

HTML. Hypertext markup language.

I

image. (1) A single page of information; the result of scanning, or digitizing, a single sheet of paper. (2) An electronic representation of a picture produced by means of sensing light, sound, electron radiation, or other emanations from the picture or reflected by the picture. An image can also be generated directly by software without reference to an existing picture. See also *page image*.

image data. Rectangular arrays of raster information that define an image. Image data is often created originally by a scanning process.

image host. The system where scanned and imported documents are permanently stored. See also *optical library subsystem*.

Image Object Content Architecture (IOCA). A structured collection of constructs used to interchange and present images.

image spool file. A file that contains sorted, merged, and completed print records ready for print.

image workstation. A programmable workstation that can perform image functions.

import data file. An AS/400 database file that contains data for one or more documents. Using data in the file, Workfolder Application Facility creates documents that can be stored and indexed just like scanned documents.

importing. A process by which documents are input into AS/400 using files rather than the scanning process. Imported documents can be stored in Workfolder Application Facility on DASD and optical, and displayed and printed, in the same manner as scanned documents.

import page ID profile. A file that contains the form overlay and fields specified for each page ID defined for a document type whose data type is 01.

inbound. Pertaining to communication flowing in a direction towards the application program from external sources, such as a transmission from a terminal to the application program. Contrast with *outbound*.

index. To associate a document with a case or identifier.

index class. A category for storing and retrieving objects, consisting of a named set of attributes known as *key fields*. When you create an item in VisualInfo for AS/400, your application must assign an index class and supply the key field values required by that class. An index class identifies the automatic processing requirements and storage requirements for an object.

indexing. The three-step process consisting of viewing a document, specifying an identifier for the document, and before creating a new case, or matching the document with an existing case.

instance. In work management, an occurrence of a work package within a process. If the process consists of parallel routes, multiple instances of a work package exist.

IOCA. Image Object Content Architecture.

item. (1) Set of attributes and objects—one or more files containing image data, annotations, notes, or other content—that together represent a physical document, such as an insurance claim or a folder.

See also *document*. (2) The smallest unit of information that the library server administers. An item can be a folder, document, workbasket, or workflow. Referred to as an *object* outside of library server functions.

J

journal. A special-purpose file or data set that can be used to provide a record of operator and system actions used to recover data and to identify operator actions that resulted in a problem.

journaling. (1) The process of recording changes made in a physical file member in a journal. Journaling allows you to reconstruct a physical member by applying the changes in the journal to a saved version of the physical file member. (2) The process of recording information sequentially in a database.

K

key field. An attribute of an item that represents a type of information about that item. For example, a customer data item might have key fields for the customer's name and social security number.

keyword. A name or symbol that identifies a parameter.

keyword field. A field enabled for input that provides data for a single keyword that is defined for a file cabinet.

keyword value. The input specified in the field for each keyword.

L

LAN. Local area network.

language profile. A control file used in Workfolder Application Facility to define country-specific parameters, such as time and date formats.

last in first out (LIFO). A queueing technique in which the next item to be retrieved is the item most recently placed in the queue.

library server. The component of VisualInfo for AS/400 that contains index information for the items stored on one or more *object servers*.

LIFO (last in, first out). A queueing technique in which the next item to be retrieved is the item most recently placed in the queue.

local area network (LAN). A computer network located on a user's premises within a limited geographical area.

LU 6.2. In Systems Network Architecture (SNA), a type of session between two application programs in a distributed processing environment, using the SNA character string or a structured-field data stream; for example, an application program using CICS communication with an AS/400 application.

M

Machine-Generated Data Structure (MGDS). (1) An IBM structured data format protocol for passing OCR (Optical Character Recognition) data among various applications. When workstations use OCR facility to create coded data from scanned images, those coded data are formatted into MGDS and passed to other applications for further processing. (2) Data extracted from an image and put into generalized data stream (GDS) format.

magnetic storage. A storage device that uses the magnetic properties of certain materials.

magnetic tape. A tape with a magnetizable layer on which data can be stored.

magnetic tape device. A device for reading or writing data from or to magnetic tape.

masking. The action of obscuring part of the image of a document so that it is not visible to the viewer.

MGDS. Machine-Generated Data Structure.

Mixed Object: Document Content Architecture (MO:DCA). An IBM architecture developed to allow the interchange of object data among applications within the interchange environment and among environments.

Mixed Object: Document Content Architecture-Presentation (MO:DCA-P). A subset architecture of MO:DCA that is used as an envelope to contain documents that are sent to the ImagePlus workstation for displaying or printing.

MO:DCA. Mixed Object: Document Content Architecture.

MO:DCA-P. Mixed Object: Document Content Architecture-Presentation.

MRI. Machine-readable information.

N

national language support (NLS). The modification or conversion of a United States English product to conform to the requirements of another language or country. This can include enabling or retrofitting of a product and the translation of nomenclature, MRI, or product documents.

network. An arrangement of programs and devices connected for sending and receiving information.

network table file. A text file created during installation that contains the system-specific configuration information for each node for each VisualInfo for AS/400 server. Each server must have a network table file that identifies it. The name of the network table is always FRNOLNT.TBL.

new case. A case that was entered and indexed, assigned to a queue, and has not been previously pended, unqueued, assigned to a process, or assigned to a workbasket. A case with this status can be accessed through the Work any case, Work queued case, or Review any case menu options.

NLS. National language support.

not queued case. A case that was entered and indexed, and is not assigned to a queue. A case with this status can be accessed through the Work any case or Review any case menu options.

O

object. (1) An item upon which actions are performed. (2) A collection of data referred to by a single name. (3) The smallest unit within the system. For ImagePlus systems, this is typically a single-image document. (4) Any binary data entity stored on an object server. In the VisualInfo for AS/400 data model, *object* specifically refers to a document's contents or parts.

object authority. The right to use or control an object.

object directory. A control file used in Workfolder Application Facility to identify AS/400 object directories used for image document storage.

object server. The component of IBM ImagePlus VisualInfo for AS/400 that physically stores the objects or information that client applications store and access.

OCR. Optical character recognition.

operator. The person who handles daily system administrative tasks.

optical. Pertaining to optical storage.

optical cartridge. A storage device that consists of an optical disk in a protective housing. See also *cartridge*.

optical character recognition (OCR). Character recognition that uses optical means to identify graphic characters.

optical disk. A disk that contains digital data readable by optical techniques. Synonymous with digital optical disk.

optical drive. The mechanism used to seek, read, or write data on an optical disk. An optical drive may reside in an optical library or as a stand-alone unit.

optical drive profile. A control file used in Workfolder Application Facility to define the optical controller used for the optical storage of documents.

optical libraries. Software used to store image data on optical platters. Only direct-attached optical systems contain optical libraries.

optical library subsystem. The hardware and software that provides the long-term storage of the image data. See also *image host*.

Optical Storage Support. Software that supports communication between stand-alone optical disk drives, the optical library, and VisualInfo for AS/400 and Workfolder Application Facility. The software runs on the System/36 5363 unit serving as the optical controller.

optical system profile. A file used to define the optical controller used for the optical storage of documents.

optical systems. Hardware used to store image data on optical platters. Only direct-attach optical systems contain optical libraries.

optical volume. One side of a double-sided optical disk containing optically stored data.

OS/2. Operating System/2.

OS/400. Operating System/400.

outbound. Pertaining to a transmission from the application program to a device. Contrast with *inbound*.

output class. A unique name assigned to a specific time frame when faxes are eligible to be transmitted.

output profile. A file that defines the content of each output form.

override. A parameter or value that replaces a previous parameter or value.

P

page. A single physical medium; for example, an 8.5-inch by 11-inch piece of paper.

page image. The electronic representation of a single physical page. The bounds of a page image are determined by the electromechanical characteristics of the scanning equipment, along with the image capture application specifications in the receiving data processing system.

page scan. The electromechanical process of scanning a physical page (paper) to create a bit image of the page.

pan. Progressively translating an entire display image to give the visual impression of lateral movement of the image.

PDF. Portable document format.

pend. To suspend a case while awaiting additional information or action, such as a particular document type or date.

pending case. A case that was pending through casework, waiting for more information. A pending case is not assigned to a queue. A case with this status can be accessed through the Work any case or Review any case menu options.

pending. Awaiting further information or action on a case.

platter. See *optical disk*.

prefix. (1) A code dialed by a caller before being connected. (2) A code at the beginning of a message or record.

Presentation Text Object Content Architecture (PTOCA). An architecture developed to allow the interchange of presentation text data.

primary processor. In a group of processing units, the main processing unit and its internal storage through which all other units communicate.

printer workstation. A display workstation equipped with a printer.

priority. A rank assigned to a task that determines its precedence in receiving system resources.

privilege. An authorization for a user to either access or perform certain tasks on objects stored in VisuallInfo for AS/400. The system administrator assigns privileges.

privilege set. (1) In VisuallInfo for AS/400, collection of *privileges* for working with system components and functions. The system administrator assigns privilege sets to users (user IDs) and user groups. (2) In the work management system in Workfolder Application Facility, an approved list of the actions, defined by a supervisor, that a user can perform on work packages. An privilege set defines options, such as printing or displaying work packages, and the function keys that are available for use.

process item. Item used as a building block in a work process.

profile. A file that governs the categories of work performed and the types of users recognized by the system.

program temporary fix (PTF). A temporary solution or bypass of a problem diagnosed by IBM as resulting from a defect in a current unaltered release of the program.

PTF. Program temporary fix.

PTOCA. Presentation Text Object Content Architecture.

Q

queue. A line or list of items waiting to be processed; for example, cases to be worked or messages to be displayed.

queue ID profile. A file that contains one entry for each active case. Each case is indexed by queue ID, queue type, and creation date and time.

R

reindexing. The process of indexing documents that were previously indexed incorrectly. This process is the same as the indexing process.

render. To take data that is not typically image-oriented and depict or display it as an image. In VisuallInfo for AS/400, you can render word-processing documents as images for display purposes.

resolution. In computer graphics, a measure of the sharpness of the image, expressed as the number of lines and columns on the display screen or the number of pels per unit of area.

rotate. A function of the document display window and the scan document display window. The orientation depends on the option selected.

route. In work management, a set of steps that move work between workbaskets, collection points, and decision points.

S

SBCS. Single-byte character set.

scan overlap. The process by which a document is scanned while a previously scanned document is stored on DASD.

scanner. A device that examines a spatial pattern one part after another and generates analog or digital signals corresponding to the pattern. (I)

scanner workstation. A display workstation equipped with a scanner.

scanning. A physical process that enters documents into an ImagePlus workstation. After a document has been scanned, it can be stored permanently.

scanning and batch indexing. An efficient scanning and indexing option that overlaps the scanning of a document with the storing of another document, while an indexing process operates automatically in the background.

search criteria. In VisuallInfo for AS/400, the text string used to represent the logical search to be performed on the library server.

secondary processor. In a group of processing units, any processing unit other than the primary unit.

server. On a local area network, a data station that provides facilities to other data stations; for example, a file server, a print server, a mail server.

side by side. A function on the document display window that displays two pages of a multipage document next to each other.

single-byte character set (SBCS). A set of characters in which each character occupies one byte.

slot. (1) A position in a device used for removable storage media. (2) A space in an optical library where an optical cartridge is stored. See *optical cartridge*.

SMS. System-managed storage.

spool file. A file that holds output data waiting to be printed or input data waiting to be processed by a program. Workfolder Application Facility can convert a spool file to an import data file.

spool writer. The part of the System Support Program that prints output saved in the spool file.

staging. The process of moving a stored object from an offline or low-priority device back to an online or higher priority device, usually on demand of the system or on request of a user. When a user requests an object stored in permanent storage, a working copy is written to the *staging area*.

stand-alone. Pertaining to an operation that is independent of any other device, program, or system.

step number. In work management, the numbers that specify the order in which route commands are processed. Each step number must have an associated command telling the route what action to take.

storage. The action of placing data into a storage device.

storage class. A storage class, in combination with an optical system identifier, defines the set of optical volumes upon which documents can be stored. Documents with the same storage class and optical system ID are stored on the same optical volume.

storage method. In Workfolder Application Facility, a means of grouping documents together for storage to an optical disk. Workfolder Application Facility provides the

following storage methods: file cabinet, collection, prioritized, and system assigned (optical distribution).

storage system. A generic term for storage in VisuallInfo for AS/400.

subsystem. A secondary or subordinate system, or the programming support part of a system that is usually capable of operating independently of or asynchronously with a controlling system.

suspend a case. To end case processing temporarily.

system administrator. The person who manages the ImagePlus workstation, the Optical Library Subsystem, and the departmental processor. The system administrator helps with problem determination and resolution. Synonymous with *administrator*.

system ID profile. A control file used in Workfolder Application Facility to define the roles of AS/400 processors in the system.

system-managed storage (SMS). The VisuallInfo for AS/400 approach to storage management. The system determines object placement, and automatically manages object backup, movement, space, and security.

System Support Program (SSP). A group of IBM-licensed programs that manage the running of other programs and the operation of associated devices, such as the display station and printer. The SSP also contains utility programs that perform common tasks, such as copying information from diskette to disk.

T

tape. See *magnetic tape*.

tape cartridge. See *cartridge*.

U

user. (1) Anyone requiring the services of VisuallInfo for AS/400. This term generally refers to users of client applications rather than the developers of applications, who use the VisuallInfo for AS/400 APIs. (2) In Workfolder Application Facility, the individual who performs input and case processing.

user activity file. A file that contains one entry for each user. It contains item counts and productivity statistics.

user exit. (1) A point in an IBM-supplied program at which a user exit routine may be given control. (2) A programming service provided by an IBM software product that may be requested during the processing of an application program for the service of transferring control back to the application program upon the later occurrence of a user-specified event.

user exit routine. A routine written by a user to take control at a user exit of a program supplied by IBM.

user fields. Data fields defined within the user exit programs. The user exit programs process the data and values passed into these fields.

user ID profile. A file that contains one entry for each user. The entries contain information such as processing eligibility.

V

volume. A certain portion of data, together with its data carrier, that can be handled conveniently as a unit.

W

workbasket. In work management in Workfolder Application Facility for AS/400, a container that holds work packages. Workbaskets can be used as parts of process definitions or ad-hoc routes. In VisualInfo for AS/400, a logical location within the VisualInfo for AS/400 system to which work packages can be assigned to wait for further processing.

A workbasket definition includes the rules that govern the presentation, status, and security of its contents.

workbasket privilege set. A list that specifies which options and function keys operate on indexing and work with work package panels.

work management. A system that lets an enterprise define a work process and environment to automate workflow and control business processes.

work management case. A case that was entered and indexed, and has been assigned to either a process or a workbasket. A work management case is not assigned to a queue. A case with this status can be accessed through the Review any case, Work with workbaskets, or Search for work packages menu options.

work order. The sequence of work packages in a workbasket.

work package. The work that is routed from one location to another. A work package can consist of an unindexed document, a file cabinet document, a Workfolder Application Facility case, or a user-defined collection of objects. It can also be empty, such as when you first create it and before it contains any work items. Work packages can be routed automatically by defined processes, or users can manually route work packages in an ad-hoc manner to workbaskets they specify.

work process. In work management, the series of steps, events, and rules through which a work package flows. A work process is a combination of the route, collection point, and decision point through which a predefined type of work package must progress.

workflow. A sequence of *workbaskets* that a document or folder travels through while it is being processed.

working set. A set of pages residing in the workstation, which can constitute one or more documents.

workstation. A computer processor unit, image display unit, scanners, and printers with which the user performs input, indexing, and printing.

Please Tell Us What You Think!

IBM ImagePlus
Workfolder Application Facility for AS/400
Designing a Work Process
Version 4 Release 1
Publication No. SC34-4588-00

We hope you found this book useful and informative. If you like what we've done, please let us know; if not, please tell us why. We'll use your comments to make the book better.

Please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number to receive a reply.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate, without obligation.

- To send comments by mail or fax, use the form titled "What Do You Think?" on the following page.

If you're mailing from a country other than the United States, you can give the form to the local IBM branch office or IBM representative for postage-paid handling.

To fax the form, use this number: (919) 254-0206.

- To send comments electronically, use one of the following network IDs:

IBM Mail Exchange	USIB5DNQ at IBMMAIL
Internet	KFRYE@CARVM3.VNET.IBM.COM

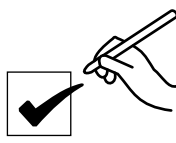
Thank you! Your comments help us make the information more useful for you.

What Do You Think?

IBM ImagePlus
 Workfolder Application Facility for AS/400
 Designing a Work Process
 Version 4 Release 1
 Publication No. SC34-4588-00

We're in business to satisfy you. If we're succeeding, please tell us; if not, let us know how we can do better.

Overall, how satisfied are you with this book?

	Very Satisfied	Satisfied	Neither Satisfied nor Dissatisfied	Dissatisfied	Very Dissatisfied	No Opinion

How satisfied are you that the information in this book is:

Accurate						
Complete						
Easy to find						
Easy to understand						
Well organized						
Applicable to your tasks						

_____ Name

_____ Address

_____ Company or Organization

_____ Phone No.

What Do You Think?
SC34-4588-00



Cut or Fold
Along Line

Fold and Tape

Please do not staple

Fold and Tape



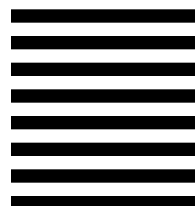
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Attn: Information Development
Department T71B Building 062
PO Box 12195
Research Triangle Park, NC 27709-2195



Fold and Tape

Please do not staple

Fold and Tape

SC34-4588-00

Cut or Fold
Along Line



Program Number: 5733-A18



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC34-4588-00

