ORACLE INSURANCE

Oracle® Documanage

Documanage Administrator's Guide

version 6.6

Part number: E14904-01

March 2009

ORACLE

THIRD PARTY SOFTWARE NOTICES

This product includes software developed by Apache Software Foundation (http://www.apache.org/).

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright © 2000-2009 The Apache Software Foundation. All rights reserved.

This product includes software distributed via the Berkeley Software Distribution (BSD) and licensed for binary distribution under the Generic BSD license.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright © 2009, Berkeley Software Distribution (BSD)

This product includes software developed by Jean-loup Gailly and Mark Adler. This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Copyright (c) 1995-2004 Jean-loup Gailly and Mark Adler

This product includes software developed by the Massachusetts Institute of Technology (MIT).

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright © 2009 MIT

This software is based in part on the work of the Independent JPEG Group (http://www.ijg.org/).

This product includes software developed by Sam Leffler of Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE

Copyright (c) 1988-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.

Publication history

First issue for Version 6.4: July 2004

Revision 1 for Version 6.4 Service Release 1: December 2004

Revision 2 for Version 6.4 Service Release 2: July 2005

Republished for Version 6.4 Service Release 3: February 2006

Revision 3 for Version 6.4 Service Release 4: November 2005

Revision 4 for Version 6.4 Service Release 5: June 2006

Revision 5 for Version 6.4 Service Release 6: July 2006

Revision 6 for Version 6.4 Service Release 6A: August 2006

Revision 7 for Version 6.5: December 2006

Revision 8 for Version 6.5: September 2007

Revision 9 for Version 6.6: March 2009

Documanage documentation road map

Administrating

Administrator's Guide	Installation Guide	Database Administrator's Guide

Programming

API help	Programmer's Guide

Workstation

Workstation Guide	Workstation help

Table of Contents

Preface

The Documanage system is comprised of server programs, including your database system and the Documanage Router and Server applications, and end-user client applications, such as the Workstation. Documanage also includes administrative applications used to manage and configure the system. This book describes those administrative applications, Documanage Administrator, Web Workflow Designer or Graphical Workflow Designer, Service Controller MMC Snap-in, Document Migration Tool and Command-line Import Tool.

Currently only the Documanage Router and Server are available for platforms other than Microsoft Windows 32-bit operating systems, and the new Web Workflow Designer is an Internet application. All other applications described in this book run only on Microsoft Windows, although they are used to administer Documanage systems where the router and server are running on other platforms. In very limited cases, a function or two described may only be available when administering router and servers running on Win32 platforms. Those cases are noted where they are described.

Installing the Administrative Applications

The Documanage administrative applications are installed as an option with the Documanage Windows services installer as described in the Documanage Installation Guide. Please refer to that guide for installation instructions.

Running the Administrative Applications

Once installed, the Administrative Applications are available in your Windows Start menu under Documanage. Documentation and help are added

to those respective sub-menus as well. To run any of the applications, click Start, navigate to the Documanage program group and choose the desired application from the list.

Using this manual

This *Administrator's Guide* describes how to use the Administrator module to set up and maintain your Documanage system. It also discusses use of the Web Workflow Designer, configuration options for non-NT security, and utilities for importing and migrating documents.

Contents

The *Administrator's Guide* contains the following sections:

- **Setting up Documanage:** Describes the procedures for setting up Cabinets and Business Tables, and defining the relationships between them.
- **Maintaining Documanage:** Describes the procedures needed to successfully maintain Documanage, including maintaining storage volumes, document Category designations, the User list and user security assignments. This chapter also discusses how to monitor and track system usage in Documanage.
- **Workflow:** Describes how to create workflow maps using both the new Web Workflow Designer and the legacy Workflow module. It also describes how to use workflow maps in a project.

- **Appendix A: Managing Heterogeneous Servers in Documanage:** A Documanage system consists of a Documanage Router and one or more Documanage Servers. The router may run on Windows NT/2000/XP, or on one of several UNIX-based operating systems, including Linux, Solaris, and AIX. A server may run on any of these same operating systems, and a single router may host servers based on more than one operating system. In Documanage the presence of multiple servers is transparent to the user. The user connects to the router and is assigned a server. Each server provides access to the same database and to the same document data store.

- **Appendix B: Documanage .ini file settings:** The Documanage .ini file settings appendix describes the sections in the poffice.ini file. The entry for each section contains a short description, a name/value table, and a sample file entry.

- **Appendix C: DmgCOMObjects .ini file:** The DmgComObjects.ini file adds one or more menu items to the Documanage Workstation Document Menu.

- **Appendix D: Using the Documanage Service Controller:** The Documanage Service Controller allows you to manage the Documanage Router and all of the Servers. To facilitate remote administration, this can be performed by any account with sufficient privileges, on any machine with network connections to the Router and Server(s).

- **Appendix E: VLAM Storage:** VLAM, an acronym for Virtual Library Access Method, is a proprietary Skywire for Oracle storage layer that uses the Virtual Storage Access Method (VSAM) to store documents and their annotations on a mainframe running the MVS (Multiple Virtual Storage) operating system.

- **Appendix F: EMC Centera Settings:** The Documanage EMC Centera API Integration product described in this appendix is available under a separate license from Oracle.

- **Appendix G: Snaplock settings:** Network Appliance Settings for Documanage.

- **Appendix H: LDAP security**: Lightweight Directory Access Protocol (LDAP) service usage with Documanage.

Conventions

This guide provides consistent typographic conventions and keyboard formats to help you locate and interpret information easily. These conventions are provided below.

Convention	Description
Italics	Command, dialog, icon, and field names
San serif font	Directory, folder, and file names
1 **Numbered lists**	Provide step-by-step procedures for performing an action
◆ Bulleted lists	Provide grouped information, not procedural steps

Related documents

In addition to this manual, the following related publication(s) are also available for Documanage:

- Installation Guide
- Using Documanage Workstation
- Database Administrator's Guide
- Dmg API online help
- Programmer's Guide

Suggestions

We welcome your comments, suggestions, and concerns about this manual or any Oracle Software publication.

Send your comments to:

Skywire's Documanage for Oracle Technical Documentation
3353 Peachtree Road NE, Ste 800
Atlanta, Georgia 30326

Setting Up Documanage

One feature that sets Documanage above the competition is its innovative *Powermapping* technology. Powermapping, a feature unique to Documanage, allows organizations to use their existing line-of-business (LOB) databases for the access and retrieval of data. Therefore, installing Documanage requires no additional programming.

In this chapter

The information in this chapter will help you answer the following questions:

- How can I PowerMap Cabinets to my existing data?
- How can I create new Cabinets?
- How do I set up Business Tables?
- How do I set up relationships between Cabinets and tables?

Overview

The chapter is broken into the following sections:

- **Setting Up Cabinets**

  This section describes the differences between Powermapping to an existing database and Powermapping from scratch.

- **Powermapping to a Database**

  This sections covers how to map to a database.

- **Creating New Cabinets**

  This sections describes creating cabinets, adding and deleting tables, creating single level, multi level, and consolidated cabinets, and releasing a cabinet.

- **Setting up Business Tables**

  This section describes business tables and covers viewing Business Table data, adding and deleting indexes, and editing index records.

- **Defining Relationships**

  This section details how to define and delete relationships between the DMANAGE “housekeeping”database and your “business” database.

Setting up Cabinets

One of the first tasks facing the Documanage Administrator is the creation of Cabinets where documents can be accessed. Cabinets are a graphic representation of data in LOB table: users will see a Cabinet icon that contains folders representing data rows in the original database tables. There are two methods by which Cabinets can be created depending on the existing database structure. No matter which method is selected, Cabinets will offer the same functionality.

- **Powermapping (using existing data):** A Documanage Administrator can map to the database table that is used as the index data. He can also customize how this data from the mapped table will be used to index and organize documents. This is the most efficient method of creating a cabinet.

- **From scratch:** A Documanage Administrator can define the cabinet and the index structure to be used for indexing and organizing documents. He can also key in the appropriate indexes.

Creating Cabinets by Powermapping

The preferred method in creating Cabinets is through the use of Skywire's Documanage for Oracle *Powermapping* feature. Powermapping, a key feature of the Documanage model, works by mapping to an organization's existing data residing in a well-defined database. Powermapping eliminates any data redundancy while avoiding prolonged data-entry time and costly programming that may be required.

For example, suppose your company's Accounting department has a database listing all current vendors. Documanage can quickly and efficiently PowerMap to this existing data. Once this database has been PowerMapped, all the data and data structures are available for use as the basis of a new Cabinet. There is no limit to the number of databases that you can PowerMap.

Creating Cabinets from Scratch

An alternate method in Cabinet creation is to create a Cabinet from "scratch." Using this method, a Documanage Administrator creates a Cabinet by first naming it and then defining all indexes that will be used to access it. A Documanage Administrator would have to create a Cabinet from scratch

when, for example, there are no databases to which to map or the existing databases are inadequate for proper document management.

Specify Cabinet on Startup

You can specify what cabinet will open when a user starts the Documanage Workstation. For example, you might have users that need access to only one cabinet. It simplifies their process to have that one cabinet open on startup. Also, if you have users that are unfamiliar with the system, it might be easiest to specify a cabinet to open at startup so they don't have to navigate through the system.

To force a specific cabinet to open you can edit the registry as follows:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Docucorp
International\DocuManage\Client]
"OnStartCabinet"="<Replace with Start Cabinet>"
```

Powermapping to a database

Since Powermapping technology works with most installed ODBC databases, indexes can be drawn from most databases currently in place; therefore, indexes will be the same in the Documanage cabinet as they are in the existing database. Documanage uses ODBC to connect to most databases.

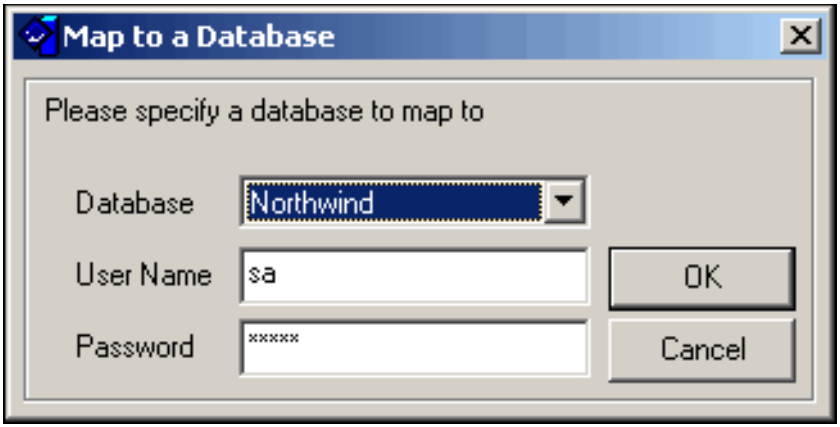
For example, a company's Purchasing document indexes can be drawn from a company's Oracle database, while its Product Development document databases can be drawn from their SQL Server database. When mapping to a database, you can map either to tables or views. Mapping to a table maps to one specific table, while mapping to a view maps you to two or more tables.

NOTE: When Powermapping, Documanage ensures that unique Cabinet names are applied to each table. If a Documanage Administrator attempts to name multiple Cabinets the same name, Documanage brings the repeat to your attention. At that time, change the duplicate Cabinet name to a unique one.

To map to a database

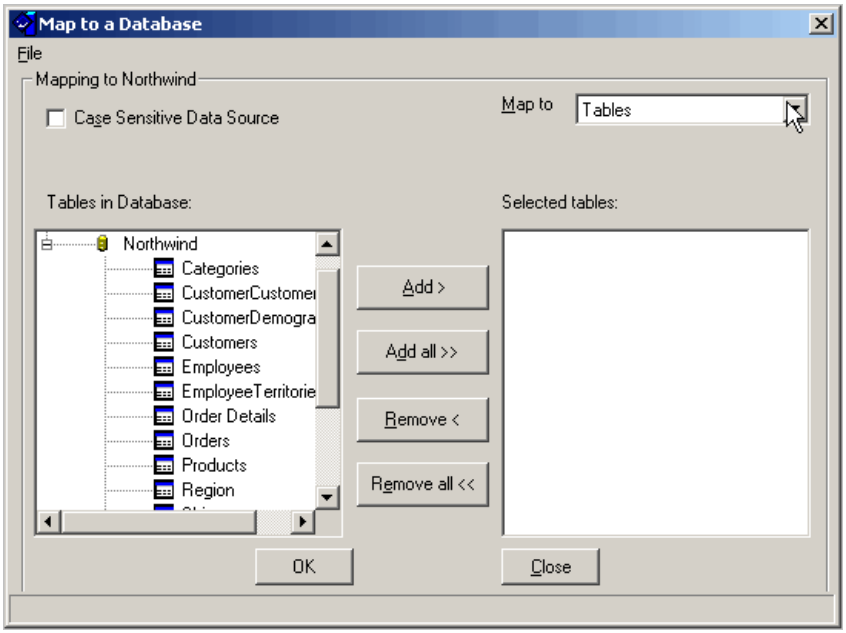
1. **Open the Documanage Administrator.**
2. **Click the *Map to a Database* button from the main Documanage Administrator dialog.**

   The Map to a Database login dialog appears.



3. **Select the appropriate database from the *Database* list.**
4. **Enter the *User Name* and *Password* for the database and click *OK*. This will be the user name and password used in the ODBC configuration.**

The Map to a Database dialog appears.



5 **Select either *Tables* or *Views from the Map to* drop-down list depending on what you want to map to.**

If you select *Tables*, the tables will be listed in the *Tables in Database* box.

If you select *Views*, the views will be listed in the *Tables in Database* box.

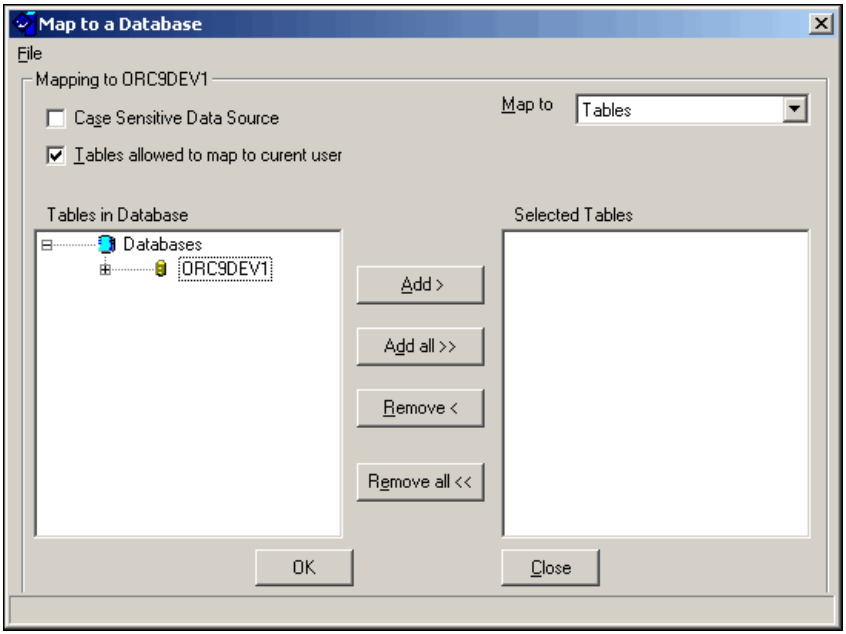
6 **Select *Case Sensitive Data Source* if necessary.**

When selected, this option omits the UCase function from the query.

NOTE: Certain data sources, such as older versions of DB2 on MVS, do not support case-insensitive searches because they lack a UCase or Upper function. By default, queries generated by the Documanage

Workstation's Query By Example dialog make use of this function, and they return an error when run against a data source which does not support the problem function. You can keep users from generating this error by using a new checkbox on the Map Database dialog.

If you are mapping to tables in an Oracle database management system, a *Tables allowed to map to current user* checkbox appears in the Map to a Database dialog. Selecting this checkbox only allows databases that you can select to display in the *Tables in Database* window.

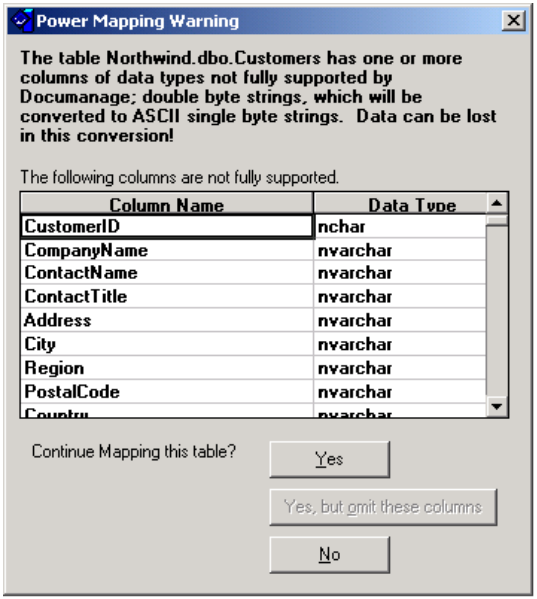


7 **Select the table you want to map from the *Tables in Database* list and click *Add* > .**

The selected table displays in the *Selected Tables* list. Click *Add all* >> to select all the tables. You can deselect tables one at a time by clicking *Remove* < or all of them by clicking *Remove all* <<.

8 **Click *OK* when all desired tables have been selected for mapping.**

If you are mapping a table that has an unsupported data type, you will see a dialog displaying the Column Name and Data type and asking what you wish to do appears.



- Continue mapping tables
- Continue mapping, but omit unsupported columns
- Stop mapping

9 **Click *OK* to close the confirmation dialog.**

All tables that were mapped are now available for document-management use within Documanage.

NOTE: If a message appears stating that the table has been previously mapped, simply click *OK* to close the message and continue. A table that has been previously mapped to Documanage, and then removed from the Documanage system, will retain a *PowerKey*, and that is why this message may appear. It is ok to re-map the table.

If you used Powermapping to link to an organization's databases, the next recommended step in creating a Documanage system is Cabinet creation. Powermapping does not automatically create Cabinets.

Creating new Cabinets

A **cabinet** is a basic unit of organization containing folders and documents that group information for a user. Documanage Administrators create cabinets to provide functional structures of information. Documanage users select cabinets by name to perform their document management operations.

Cabinets can contain levels. A **cabinet level** is a layer within a cabinet and is created by the Administrator when cabinets are defined. Cabinet levels are related hierarchically and each level references a set of one or more LOB tables.

A cabinet level contains a sequential level number, a set of one or more LOB table names, and an SQL relationship to the immediately preceding level's table set (if a preceding level exists).

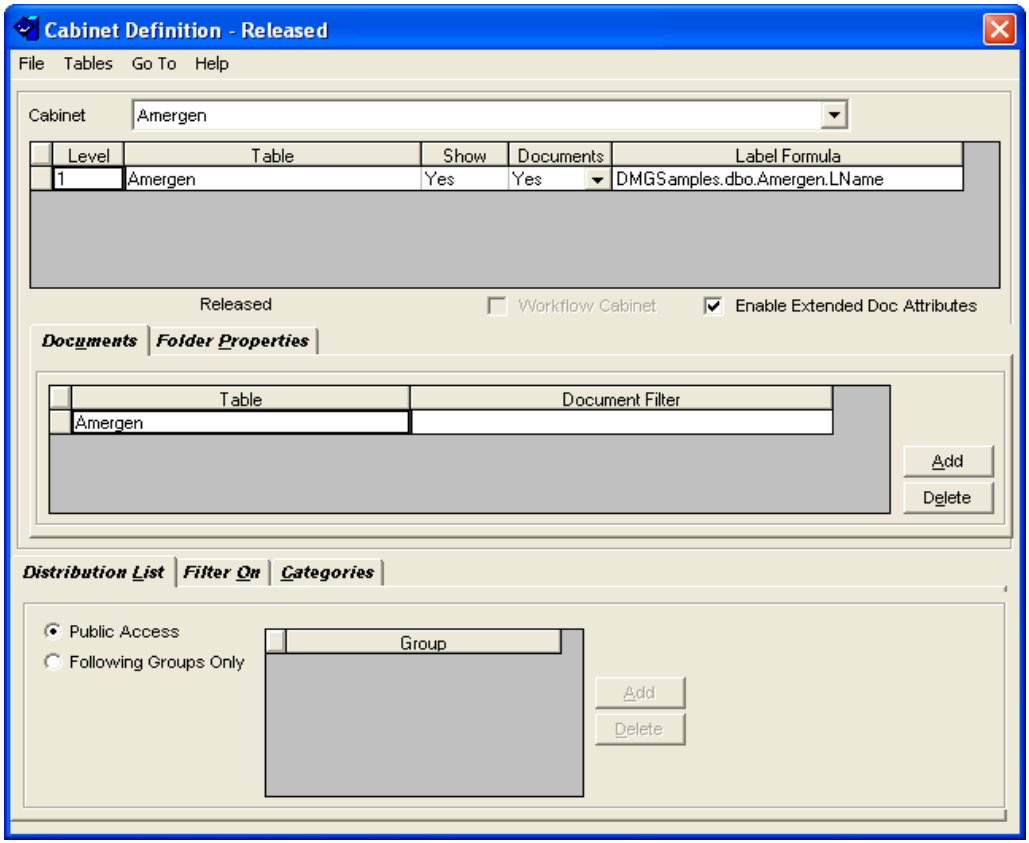
Cabinets that contain one level are sometimes referred to as Single Level Cabinets. Cabinets that contain more than one level are often called Multi-Level Cabinets.

Cabinet levels can be consolidated. **Consolidation** groups documents from multiple LOB tables and organizes the documents in one folder level so the documents can be viewed together.

For example, you might have one folder for the LOB table *Vehicle Types* and one folder for LOB table *Manufacturers*. Once these tables are defined in a relationship, a consolidated level could show all vehicle types and their manufacturers in one folder.

Use the *Cabinet Definition* dialog to create new cabinets. From this dialog you can view folders in a cabinet, folder properties and documents, and

groups that have access to the selected cabinet. You can also specify a filter on which the end user can search for documents.



Cabinet Definition Dialog Options

Field	Description
Level	The level the table will appear in the Cabinet
Table	The table name
Show	Whether or not that table will show in the Client in that cabinet
Documents	Whether or not documents will show in the cabinet's folders
Label formula	Defines the default folder label that will appear within the Documanage Client. This can be a combination of text and fully qualified table columns as defined by the database you are using.
Documents tab:	
Table	The table name
Document filter	Allows you to set both document and folder filters using SQL Where clauses
Properties tab:	
Property name	The name of the field
Display	Whether or not that field will appear in the Client
Editable	Whether or not that field is editable
Required	Whether or not that field is required to be completed in the Client
Distribution List tab:	
Public Access	Provides access to all groups in Documanage. Security on tables will also be implemented on all cabinets
Following Groups Only	Provides access to selected groups in Documanage. Security on tables will also be implemented on all cabinets
Group	Only users belonging to the selected groups can access the cabinet

Field	Description
Filter On tab:	
Filter text box	Allows you to enter a SQL WHERE clause that filters cabinet folders
Variable (See Note)	Opening a cabinet prompts you to enter a value for the variable defined in the Filter text box. The Workstation filters folders in the cabinet which match the value of the variable before the cabinet opens.
Order By	Overrides default sorting
Type	The variable data type
Categories:	
Include All/Exclude All	Allows selection (check box) of Document Categories that are to be EXCLUDED from use with this Cabinet. Users will not be able to use a specified, or checked, Document Category with this particular Cabinet. By default, all Categories are available for use with all Cabinets.

NOTE: The Variable must be enclosed in % symbols. The query shown here filters the `CalXPower` table in the `DMGSamples` database using its `Service_State` column:
`DMGSamples.dbo.CalXPower.Service_State LIKE %State_Code%.`
`%State_Code%` would be entered into the Variable box. The entire statement would be entered in the Filter text box.
As you open the cabinet, the Workstation prompts you to enter a value for the `State_Code` variable. The filtered cabinet only shows folders containing the value of `State_Code` in the `Service_State` column.
Alternately, you can choose from a list of values for the variable that

have been entered in the poffice.ini file. Refer to “Cabinet Filters” on page 241.

Creating a single-level cabinet

The most basic type of cabinet is the Documanage single-level cabinet, which contains only one table. This type of table is created through the *Cabinet Definition* dialog.

To create a single-level cabinet

1. **Click the *Cabinets* button in the main Documanage Administrator dialog.**

2. **Select *New* from the File menu.**

3. **Type a new cabinet name in the *New Cabinet* dialog and click *OK*.**

CAUTION: Cabinet names cannot be all numeric: they must contain at least one alphabetic character.

4. **Select a table from the *Table* list.**

   Adding one table creates a single level cabinet.

5. **Select whether or not users are able to add Documents to the table in the *Documents* column.**

   If you select **`Yes`**, the user can view and add documents to this table. If you select **`No`**, the user can view documents in this folder, but cannot add any documents.

6. **Edit the Label Formula, if necessary.**

This is the label for the table that will display in the Client. Folder labels must be entered in valid label format which is `databasename.dbo.tablename.primarykey`.

7 **Select *Enable Extended Doc Properties* if you want to enable extended document properties for this table.**

Extended document properties are created on Document Categories|Extended Doc Properties. For more information see “Adding Extended Document Properties” on page 103.

8 **Go to the *Documents* tab, add tables, and define the document filter.**

9 **Go to the *Folder Properties* tab. For each Database Field Name, select whether the field will *Display*, is *Editable*, or is *Required* in Documanage Workstation.**

10 **Set the *Distribution List, Filter On,* and *Categories* properties.**

11 ***Save* and *Exit.* Upon Saving you may be prompted to Refresh the connection to the Documanage Server. Supply the ODBC username and password here.**

Creating a multilevel cabinet

A multilevel cabinet contains two or more tables. This type of cabinet is created through the *Cabinet Definition* dialog.

To create a multilevel cabinet

1 **Click the *Cabinets* button in the main Documanage Administrator dialog.**

2 **Select *New* from the *File* menu or select an existing cabinet from the *Cabinet* list.**

3 **Type a new cabinet name in the *New Cabinet* dialog and click *OK*, if you are creating a new cabinet.**

CAUTION: Cabinet names cannot be all numeric: they must contain at least one alphabetic character.

4 **Select a table from the *Table* list.**

Selecting one table creates a single level cabinet. Adding two or more tables creates a multi-level cabinet.

5 **Select *Add Table* from the *Tables* menu.**

A second level displays.

6 **Select a table from the *Table* list.**

Add as many tables as necessary to the multi-level cabinet.

If a relationship does not already exist between the selected tables, the *Define Relationship* dialog opens and you must define a relationship between the database tables to continue. For more information see *"Defining Relationships" on page 30*.

7 **Select whether or not you want to *Show* the table.**

After adding two or more levels, the *Show* field becomes active for each level except the last.

If you select **Yes**, the user will be able to see this table displayed as a folder in the Documanage Client. If you select **No**, the user will not be able to see this table in the Documanage Client.

8 **Select whether or not users are able to add Documents to the table.**

If you select **Yes**, the user can view and add documents to this table. If you select **No**, the user can view documents in this folder, but cannot add any documents.

9 **Edit the *Label Formula*, if necessary.**

This is the label for the table that will display in the Client. Folder labels must be entered in valid label format which is `databasename.dbo.tablename.primarykey`.

10 **Select *Enable Extended Doc Properties* if you want to enable extended document properties for this cabinet.**

Extended document properties are created on Document Categories|Extended Doc Properties. For more information see “Adding Extended Document Properties” on page 103.

11 **Go to the *Documents* tab, add tables, and define the document filter.**

12 **Go to the *Folder Properties* tab. For each Database Field Name, select whether the field will *Display*, is *Editable*, or is *Required* in Documanage Workstation.**

13 **Set the *Distribution List, Filter On,* and *Categories* properties.**

14 ***Save* and *Exit.* Upon Saving you may be prompted to Refresh the connection to the Documanage Server. Supply the ODBC username and password here.**

Creating a consolidated cabinet

A consolidated cabinet contains documents from multiple tables at the same cabinet level.

To create a consolidated cabinet

1. **Click the *Relationships* button from the Documanage Administrator dialog, define a relationship, save and close the form.**

   See “Defining Relationships” for directions on completing the dialog.

2. **Select the *Cabinets* button in the main Documanage Administrator dialog.**

3. **Select *New* from the *File* menu.**

4. **Type a new cabinet name in the *New Cabinet* dialog.**

CAUTION: Cabinet names cannot be all numeric, they must contain at least one alphabetic character.

5. **Select the desired table for the cabinet from the *Table* drop-down list.**

6. **Select *Add* on the Document tab and add a new table. Repeat as necessary**

   See “Adding and deleting tables” for details.

7. ***Save* and *Exit* the form.**

Adding and deleting tables

To add a table

1. **Select the Cabinet to which you want to add tables from the *Cabinet* drop-down list.**

2. **Select the *Add Tables* option from the Tables menu or press [Alt+T+A].**

3. **Select the table you want to add from the *Table* drop-down list.**

   The Define Relationship dialog appears when you add a second table. Complete the form. See “Defining Relationships” for directions on completing the dialog.

4. **Select the whether or not you want to *Show* the table.**

   If you select Yes, the user will be able to see this table displayed as a folder in the Documanage Client. If you select No, the user will not be able to see this table in the Documanage Client.

   All tables can be set to show or not show except the last one listed. The last table listed will always be visible in the client. For example, if 5 tables are listed, you can hide the first 4. The 5th one will display in the Client. If 10 tables are listed, you can hide the first 9, the 10th one will display in the Client.

5. **Select whether or not you want users to be able to add Documents to the table.**

   If you select **`Yes`**, the user can view and add documents to this table. If you select **`No`**, the user can view documents in this folder, but cannot add any documents.

6. **Edit the Label Formula, if necessary.**

   This is the label for the table that will display in the Client. Folder labels must be entered in valid label format: `databasename.dbo.tablename.primarykey`.

7 ***Save* and *Exit*. Upon Saving you may be prompted to Refresh the connection to the Documanage Server. Supply the ODBC username and password here.**

To delete a table

1. **Select the table you want to delete.**
2. **Select the *Delete Table* option from the Tables menu or press [Alt+T+D].**

   A confirmation box displays.
3. **Select *Yes*.**

   The table is deleted.

Editing a Database Field

You can edit database field properties on the Cabinets dialog. You can determine whether a user will be able to see a field, whether they can edit the field, and whether the user must enter information in that field.

After selecting the Cabinet you want to work with, go to the *Folder Properties* tab and select **`Yes`** or **`No`** in the *Display*, *Editable*, and *Required* field depending on the properties you want to set.

Releasing a cabinet

Before any Documanage user can access a cabinet, it must first be released using the *Cabinet Definition* dialog. Releasing a cabinet (including single-, multi-level, or consolidated cabinets) permits it to be viewed by Documanage users with the appropriate rights. Cabinets that are not released will ***not*** appear to users in the Client. Unreleased cabinets are labeled as *Not Released* and released cabinets as *Released*.

NOTE: Editing a cabinet definition in any way causes the cabinet to be *Not Released* again. Therefore, after making changes to a cabinet definition, it is necessary to resave and re-release the cabinet.

To release a cabinet

1. **Be sure that the cabinet you want to release appears in the Cabinet Definition dialog and has been saved.**

2. **Select the *Release* option from the File menu or press [Alt+F+R].**

3. **Enter the *User Name* and *Password* for the selected database.**

   The cabinet is released and available for viewing in the Documanage Client by those viewers who have the rights to do so.

Viewing, Modifying and Setting Up Business Tables

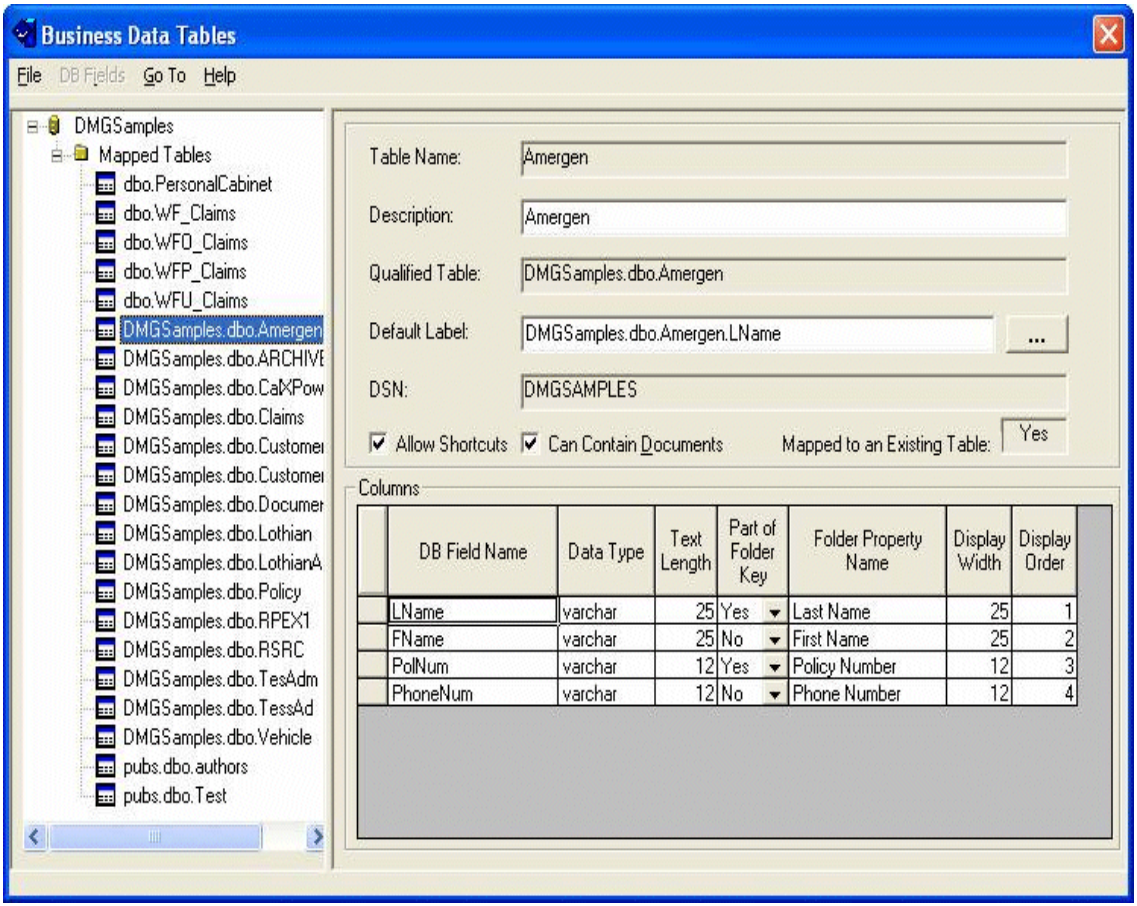
Every Documanage table has attributes associated with it. These attributes include a description of the table, its related database, data type, whether or not shortcuts to the table can be made, and so on.

To quickly view the attributes related to a database, you can use the *Business Data Tables* dialog accessed by clicking *Business Tables* in the *Documanage Administrator* dialog. You can use the *Business Data Tables* dialog to view, modify or create attributes for an existing or a new database table.

For example, to view the attributes of the Authors table, select the table icon labeled *Authors* from the list of table icons displayed in the window on the left hand side of the dialog. Authors appears in the *Table Name* field in the dialog, and the characteristics of the table columns appear in the Columns section of the dialog.

Business Data Table Dialog

The screen shot that follows shows the *Business Data Tables* dialog.



Business Data Table Dialog Fields and Options

The table that follows defines the fields in the *Business Data Tables* dialog

Field	Description
Table Name	The name of the Documanage housekeeping table you are creating or modifying, or you have selected to view
Description	A quick description of the purpose of the table. It is for the Administrator's use only and does not appear in the Client.
Default Label	Defines the default folder label that will appear in the Documanage Client. This text is from the label for a fully qualified table column as defined by the database you are using.
DSN	The data source name of the data base
Allow Shortcuts	Select *Yes* to allow shortcuts and checked-out documents to be kept in the table, or select *No* to disallow them
Can Contain Documents	Whether or not the database table can contain documents
Mapped to An Existing Table	Indicates whether the selected line of business table column field is powermapped using the Map to a Database dialog in the Administrator
DB Field Name	The name of the table field
Data Type	The data type of the table field defined by the ODBC driver.
Text Length	The length of the table field
Part of Folder Key	Every table used in Documanage requires a unique value to be assigned as part of a folder key. This is the same as the primary key for the table.
Folder Property Name	How the database field text appears in the Client (that is, the name that the Client displays)
Display Width	The width of the table fields displayed in the Client
Display Order	The order in which the table fields appear in the Client

Viewing the contents of a Business Table

1. **Click *Business Tables* in the main *Documanage Administrator* dialog.**

   The *Business Data Tables* dialog appears.

2. **In the tree displayed on the left side of the *Business Data Tables* dialog, click on the icon for the database table that you wish to view.**

   The characteristics of the selected table display in the fields on the right side of the dialog.

3. **Click *File* | *Exit*. when you are finished viewing the table.**

Modifying a Business Table

As the needs of your enterprise grow, you may need to add additional database fields to a table or update a table in Documanage that you have changed in your database. In addition to naming a new database field, you may specify other information such its data type, length and whether it will be part of the table key.

In the *Business Data Tables* dialog, you can also delete any database fields that you don't want to make available to Documanage users. Doing so keeps these documents or document categories hidden from unauthorized users.

NOTE: You can only add database fields to newly created tables.

To add a database field

Follow the steps below to add a field to a data base table.

1. **Click *Business Table* in the *Documanage Administrator* dialog.**
2. **In the tree displayed on the left side of the *Business Data Tables* dialog, click on the icon for the database table that you wish to view.**
3. **Enter a *Description* for the table.**

   This is for your future reference. It does not appear in the Client.
4. **Select *Add DB Field* from the *DB Fields* menu.**

   A new row appears in the table.
5. **Type a name in the *DB Field Name* field.**
6. **Select a data type from the *Data Type* list.**
7. **Type the length of the *DB Field Name* in the *Text Length* box.**
8. **Click the *Part of Folder Key* list and select *Yes* or *No*.**

   Selecting *Yes* makes this field a part of the folder key.

NOTE: A folder key is a set of fields that uniquely identify a Documanage folder. If two folders have the same key, Documanage outputs errors. To ensure that a folder key is unique, choose at least one key that is unique to the database among the fields that form the folder key. (If the unique key is a composite key, all its constituent fields must be selected to be part of the folder key.)

9. **Type the name that you want to represent the DB field in the *Folder Property Name* field.**

   This is the name that appears in the Documanage Workstation (Client).

10 **Enter the width of the field that displays in the Client in the *Display Width* field.**

11 **Enter the order that fields display in the Client in the *Display Order* field.**

12 **Click *File* | *Save* and *File* | *Exit*.**

To update a database field

Perform these steps if you change a field in your database and want Documanage to reflect the changes.

1 **Enter a *Folder Property Name* if you want the user of the Client to see a different name on a folder than the one displayed.**

2 **Go to the *Cabinets* dialog and delete the cabinet(s) referencing the table.**

   This does not destroy the tables in the database.

3 **Go to the *Business Tables* dialog and *delete the table* you modified in the database.**

   This does not delete the table from the customer database. Data will not be lost.

4 **PowerMap the updated tables with the new indexes.**

   See “Powermapping to a database” on page 4 for more information.

5 **Recreate the cabinets.**

   See “Creating new Cabinets” on page 9 for more information.

To delete a database field

CAUTION: Use the *Delete DB Field* option with caution, since you may not be able to immediately retrieve deleted fields.

1. **Select the *DB Field* you want to delete.**
2. **Select *Delete DB Field from the DB Fields* menu or press [Alt+I+D].**
   A confirmation message appears.
3. **Click *Yes*.**
   The *DB Field* is deleted.

Creating a new Business Table

When you create a new table from Documanage, follow these steps to add a field to the table.

1. **Click *Business Table* in the *Documanage Administrator* dialog.**
   The *Business Data Tables* dialog appears.
2. **Click *File* | *New*.**
   The boxes in the dialog go blank. A row appears in the *Columns* area.
3. **Populate the row by performing “Step 5” through “Step 11” on page 27 through page 28. Press Tab.**
   A row appears in the *Columns* area.
4. **Repeat “Step 3” until you have populated all of the rows in the table.**
5. **Type the name of the new table into the *Table Name* box.**
6. **Type a brief description of the table into the *Description* box.**

7 **Type a default folder label that will appear in the Documanage Client into the *Default Label* box.**

NOTE: To make this entry, you can also click "..." and type the label into the *Folder Label Formula* dialog.

8 **Type the Data Source Name (DSN) into the *DSN* box.**

9 **Click *File* | *Save* and *File* | *Exit*.**

Defining Relationships

A relationship creates a Cabinet consisting of two or more tables. For example, if you want to create a Cabinet that lists employees and the human resources department, you would create a relationship between the Employee and Human Resources tables. The similar value database fields in these tables would be used to create a relationship—for example, Employee ID.

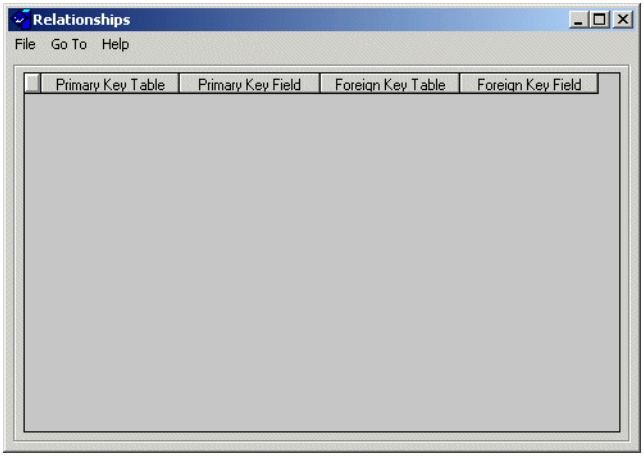
New relationships are defined through the *Define Relationship* dialog.

NOTE: You can quickly switch to the Business Tables dialog from the Relationships dialog by selecting *Business Tables* from the Go To menu as conceptually they are similar.

To define a relationship

1 **Click the *Relationships* option on the main Documanage Administrator dialog.**

The *Relationships* dialog opens.



2 **Select *File* | *New*.**

3 **Click the *One Record Table* drop-down list to select a table to be joined.**

A list of indexes for that table appears.

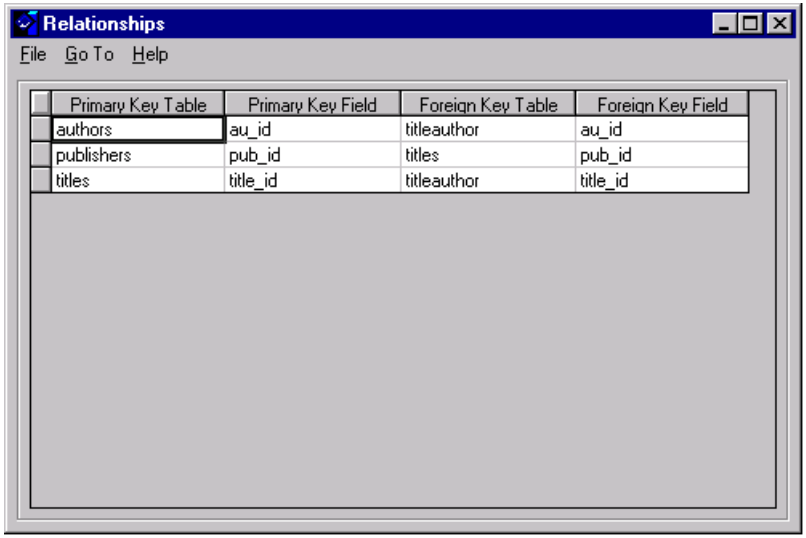
4 **Select the index by which you will join the tables.**

5 **Click the *Many Records Table* drop-down list.**

Once the table is selected, the corresponding database field will automatically fill in. If one does not fill in, you can use the drop-down menu to select a joining index.

6 **Click *OK* when finished.**

The *Relationships* dialog reappears with the new relationship shown.



7 ***Save* and *Exit* the form.**

The *Define Relationship* dialog closes.

To delete a relationship

1 **Select the table you want to delete.**

2 **Select *File|Delete* or press [Alt+F+D].**

A confirmation box appears.

3 **Click *Yes*.**

4 ***Save* and *Exit* the form.**

Processing Large Metacode Files

Documanage now handles metacode files larger than 256 Kb. This is handled through the IMC.INI settings MaxDocumentSize and MaxSeekSize.

MaxDocumentSize specifies the maximum document size to be loaded in units of Kb. The default is 256 Kb. The Printstream Library stops opening a document when it reaches the end of the document, reaches that maximum, or runs out of memory, whichever comes first. The only pages that would then be available are those that are fully described up until that point. In the latter two cases, the user will receive a warning stating the full document could not be read.

MaxSeekSize specifies 2 things. It specifies how far past the beginning of a file (in Kb) the printstream library looks for the first document boundary without finding one before deciding that there is only one document in the file. For example, if MaxSeekSize is set 1000K, the printstream library will look through 1000k of the file and if it does not find a document boundary it will decide there is only one document in the file.

MaxSeekSize also specifies how far after the place where the printstream library last stopped reading (the end of the last document it read, or MaxDocumentSize into the last document it read) that it looks for the next beginning document marker before it stops.

For example, suppose MaxDocumentSize is set to 1000K and MaxSeekSize is set to 1500K. Suppose also that a file contains two documents of 8000K and 500K respectively. In reading the first document in the file, the printstream library will stop reading the file at 1000K. If the second document is requested, the printstream library will continue to look for the next beginning document marker for 1500K beyond the point it last read. Since the printstream library does not find a beginning document marker within the 1500K, it decides there are no more documents in the file.

Keep in mind that the maximum document size that may be effectively processed on your system will be affected by the amount of memory available on your computer, the speed of your computer and these product settings described above.

Configuring Documanage Logging

This section explains how to configure Documanage logging, the component parts of a Documanage log file, and how the log file capacity is managed.

Documanage Logging Configuration

Configuration of the Documanage logging framework can be done via two methods: the poffice.ini file, and the Microsoft Management Console (MMC)

Poffice.ini File Configuration Method

The poffice.ini file contains configuration information for each of the Documanage applications. Within these sections, the system administrator defines name/value pairs that specify the logging level, detail level, log file name, log file size, startup action, and SQL logging level. In the following sections, we will be taking a look at each of these settings.

LogDisable

This entry determines whether logging is allowed or not. A value of '0' (the default) means the logging for this application is allowed. A value of '1' disables all logging from the application.

LogDetailLevel

This entry specifies which entries to the log file are actually made. Each message has a level of importance associated with it. '0' means the message is very important and should appear whenever logging is enabled. '1' means that the message is relatively important, but does not usually adversely affect the execution of the application. Finally, '2' means the message is for informational purposes only.

By modifying the LogDetailLevel value ('0' is the default), an administrator can manage the number and quality of messages that are written to the log file.

LogExtraDetail

Each message that is written to the log file contains some basic information (timestamp, thread id, message). By setting this entry to '1' ('0' is the default), some additional information will be written if available. This includes the raw error value, the module/function causing the error, and the module/function catching and processing the error.

LogFile

This entry specifies the name of the log file being written to. The Documanage Server's default log file name is "poffice.log." The Documanage Router's default log file name is "dmgrouter.log."

A second file is created from this entry (the archive file) with ".old" appended to the value.

LogMinSize

This entry has been deprecated, so administrators may safely exclude it.

LogMaxSize

In order to prevent the log file from growing too large, an administrator may specify a maximum log amount (in Kbytes). The logging framework will split this amount between the current log file ("poffice.log" for example) and the archive log file ("poffice.log.old").

The default log file maximum size is 200 Kbytes.

LogSqlLevel

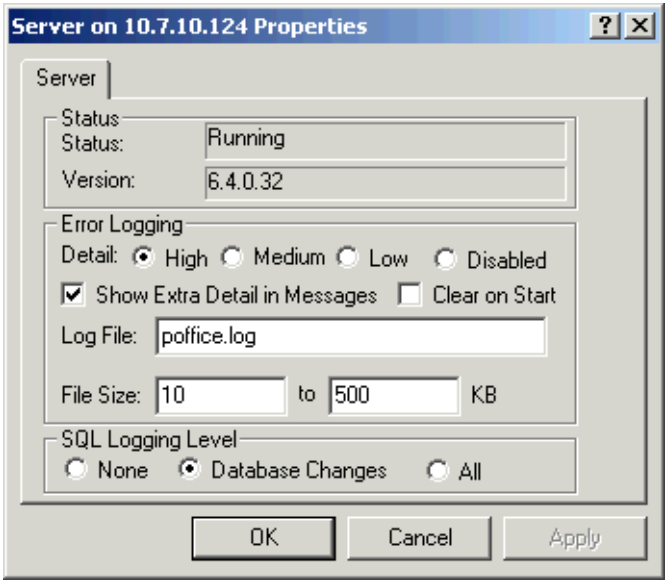
Just as determines which messages are written to the log file, LogSqlLevel determines which database messages are written to the log file. The values for this entry determine if and which messages get logged. A value of '0' (the default) specifies that no logging be done. '1' specifies that all non-select statements be logged. And lastly, '2' specifies that all statements be logged.

DeleteFileAtStartup

This entry tells the server to delete the current contents of the log when during its initialization. If the value is '0' (the default), the server will NOT truncate the data in the log file. If it is '1,' it will.

Microsoft Management Console Configuration Method

The MMC may be used to manage the Documange Server's log settings. Opening the MMC and drilling down into the explorer style tree, right click the desired server and select "Properties" from the popup menu. The following Documanage Server Properties dialog appears:



Using this dialog, a system administrator can set all the values reference in the poffice.ini section above.

Once the desired values are set, click the "Apply" button to forward them to the server and click "OK" to exit. Click "Cancel" to backout any changes that have not yet been applied.

Documanage Log File Components

This section describes the component parts of a Documanage log file entry.

The following text shows a typical log file entry:

07/08/04 11:53:03.949 [0xbb0] - [*] - Task queue thread for time 60000 starting.

A log file entry is made up of the following parts:

- Timestamp—The first section of the entry that displays the current date followed by the time (including milliseconds);
- Thread ID—The hexadecimal representation of the tread handle from where the log message originated;
- Log Level Indicator—A number of asterisks representing the message's log level: One asterisk for LogDetailLevel 2 messages, Two asterisks for LogDetailLevel 1 messages, and three asterisks for LogDetailLevel 0 messages. See the poffice.ini section above for the definitions of the LogDetailLevel levels.
- Message Text—The log message's text. This section may or may not contain extra detail depending on 1) the LogExtraDetail setting, and 2) the source of the message.

Log File Capacity Management

It is often the case that an administrator of a Documanage system wishes to limit the amount of hard drive space taken up by the Documanage log file(s). The configuration variable LogMaxSize maintains the maximum amount of disk space to be used by the logging framework.

This disk space is divided equally between the current log file and the archive log file (<current log filename>.old). These files are managed as follows:

1. **The application starts up and checks the DeleteFileAtStartup value to determine if the current log file needs to be cleared or not.**
2. **Then, when a message is received for writing, the system checks the message's level against the current logging level (as specified by LogDetailLevel), and if appropriate, forwards it for writing.**
3. **The application then checks the size of the current log file. If it is greater than or equal to 50% of the value specified by LogMaxSize (in Kbytes), any archive file is removed and the current log file is converted to the archive log file.**
4. **The message is then written to the current log file (creating a new one if necessary).**

Maintaining Documanage

Documanage System Administrators must perform various functions to successfully maintain Documanage, such as maintain storage volumes, document Category designations, the User list, and user security assignments. Documanage also provides the ability to monitor and track system use to optimize performance.

In this chapter

The information in this chapter will help you answer the following questions:

- How do I add, delete, and migrate Documanage storage volumes?
- How do I maintain Documanage document categories?
- How do I define default storage options, and monitor or track system use in Documanage?
- How do I maintain the User list and define security levels?

Maintaining storage volumes

Documanage stores documents in volumes. It supports most types of storage volumes, including magnetic and optical disks, magnetic tapes, CD-ROMs, BLOB storage in database tables, Network Attached Storage (NAS) systems, and Storage Area Network Storage (SAN) systems.

For organizations with complex storage needs, including retention and migration requirements, Documanage supports a wide range of Hierarchical Storage Management (HSM) systems. In fact, Documanage supports any storage device or HSM system that DOS recognizes as a drive.

Documanage also makes the distinction between *permanent*, *temporary* and *annotation* storage. *Permanent* storage contains the original Documanage documents while *temporary* storage contains the checked-out copies of Documanage documents. *Annotation* storage contains annotations for documents.

You can set up any number of *volumes* for a document Category, and there are two ways in which documents are stored in volumes: round-robin format (default) and sequential fill format. When using the round-robin format, each time you import a document into the Documanage system, the document is placed into the next consecutive volume for that document Category.

For example, if you are importing 4 documents into the system with the same document Category and that Category has 3 volumes, the first document is imported into Volume 1, the second into Volume 2, the third into Volume 3, and the fourth into Volume 1.

With the sequential fill format, documents with the same document Category are imported into one volume until it is full, at which point documents are imported into the next volume with sufficient space.

For example, if you are importing 100 documents and the specified Category has 2 volumes, documents are imported into Volume 1 until it is full, at which point the documents are imported into Volume 2.

You can also set the space you want to reserve on each volume for revisions. This allows you to revise a document and ensure it is placed in the same volume as the original.

Storage concerns and tasks are accomplished through the *Storage* option in the main Administrator module.

Displaying volumes in storage

If you want to make certain that all your current storage devices are represented properly, you can display the volumes listed on your Documanage Server in the *Storage* dialog on the *Volumes* tab.

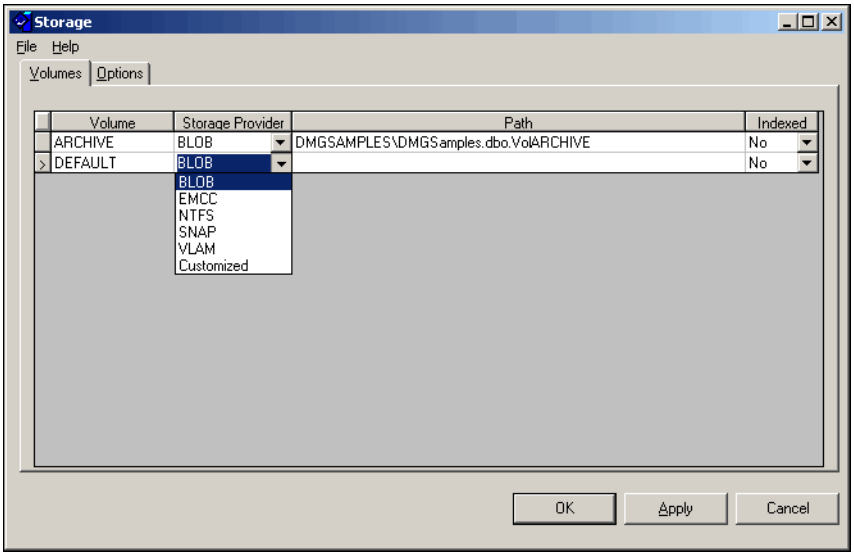
To display volumes in storage

1. **Click the *Storage* option on the main *Documanage Administrator* dialog.**
   The *Storage* dialog appears.

2. **Open the *Volumes* tab.**

   The *Volume* list contains the current volumes, and the *Storage Provider* column lists the volume types (*BLOB*, *NTFS*, *SNAP*, *EMCC*, *VLAM* or *Customized*) and the *Path* column lists the volume locations. If you are using a Volume on a Windows operating system, you can index it, which enables a full text search capability using Microsoft Index Server.

NOTE: The Microsoft Index Server must be running on the same server as the Documanage Server.



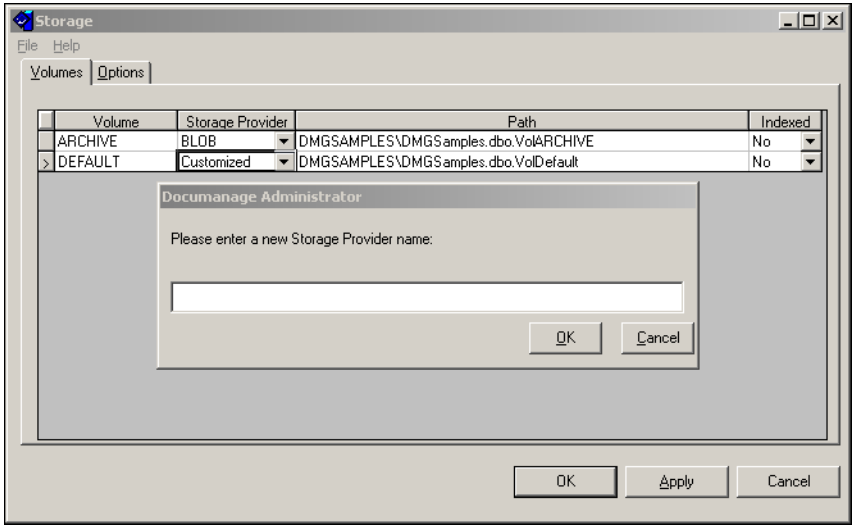
Adding a custom storage provider

You can choose from one of the storage providers in the Storage Provider list, or you can enter the name of a custom storage provider. No dialog will be provided for the path if a custom specifier is entered; you must enter the path manually.

1. **With the *Storage* dialog displayed, select *Customized* from the *Storage Provider list.***

   A *Documanage Administrator* dialog appears.

2. **Type the name of the custom storage provider into the box, then click *OK*.**



Adding a new volume

Using the Storage dialog, you can add new volumes to Documanage as your storage needs increase. To do this, you enter a volume name for the storage device and its associated path in the *Storage* dialog. You must create the storage location before inputting it into the *Path* box. You can add as many additional volumes as required.

NOTE: The procedures that follow describe how to automatically enter a path for an NTFS, SNAP or BLOB volume into the *Path* box using a browse dialog. EMCC, VLAM and Customized volumes must have their path manually typed into Path field.

To add a new NTFS or SNAP volume

NOTE: Directories must already exist before you enter them into the Path field.

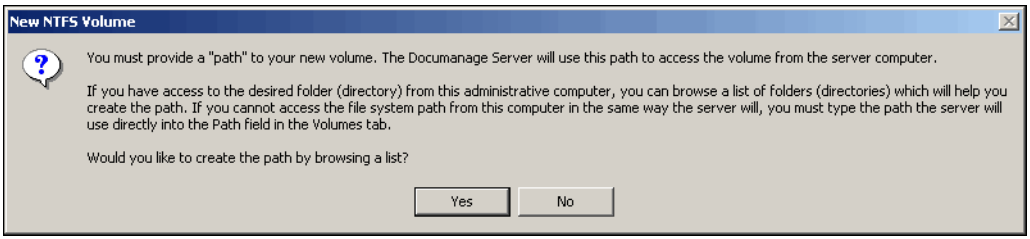
1. **With the *Storage* dialog displayed, select *New* from the *File* menu or press [Alt+F+N].**

   An insertion point appears in the first open *Volume* field.

2. **Type the desired volume name, then press the [Tab] key.**

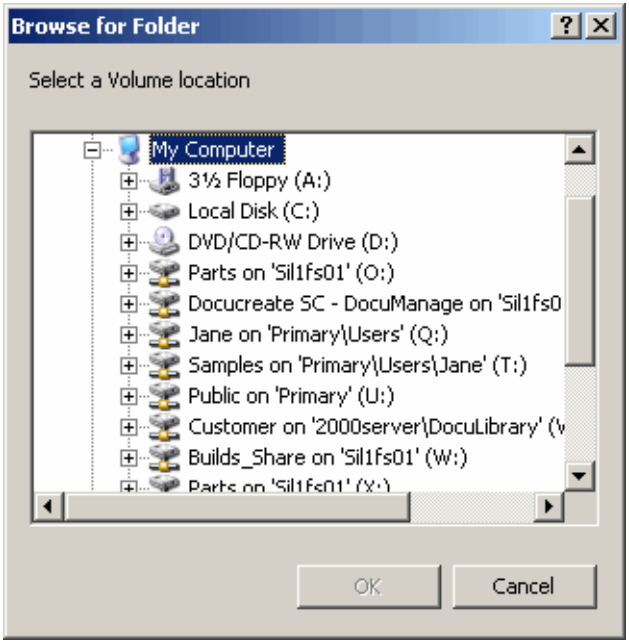
3. **Select the type of volume from the *Storage Provider* list.**

   A *New NTFS Volume or New SNAP Volume* message appears.



4. **Click *Yes* if you want to navigate to the location of the new volume rather than typing its path directly into the *Path* box.**

The *Browse for Folder* dialog appears.



5 **In the dialog, navigate to the location of the new volume. Click *OK*.**

The path to the location of the new volume appears in the *Path* field in the *Storage* dialog.

6 **Select *Yes* or *No* from the *Indexed* list in the *Storage* dialog, then press the [Tab] key.**

Selecting *Yes* enables a full text search using a Microsoft Index Server

7 **Click *OK* and *Exit*.**

To set up a BLOB storage volume

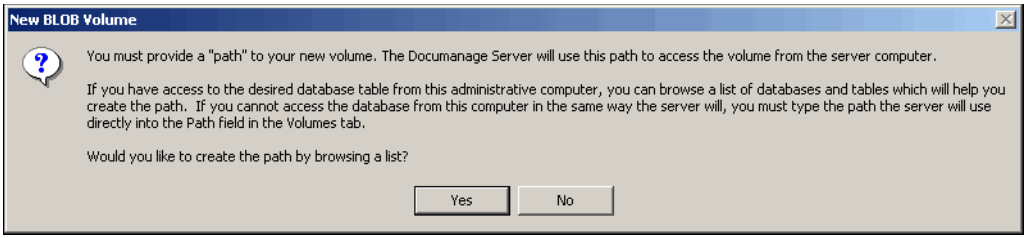
1. **From the *Storage* dialog, select *New* from the *File* menu or press [Alt+F+N].**

   An insertion point appears in the first available open *Volume* field.

2. **Type the desired volume name, then press the [Tab] key.**

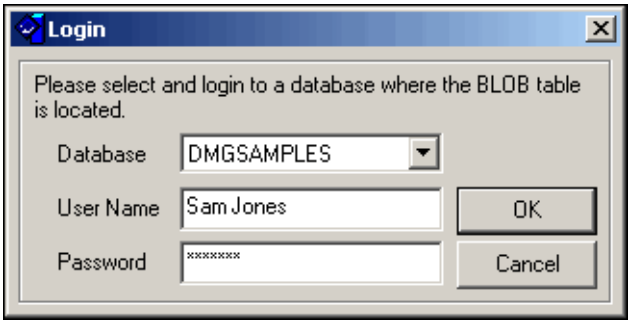
3. **Select *BLOB* from the *Storage Provider* list.**

   A *New BLOB Volume* message appears.



4. **Click *Yes* if you want to navigate to the location of the BLOB table rather than typing its path directly into the *Path* box.**

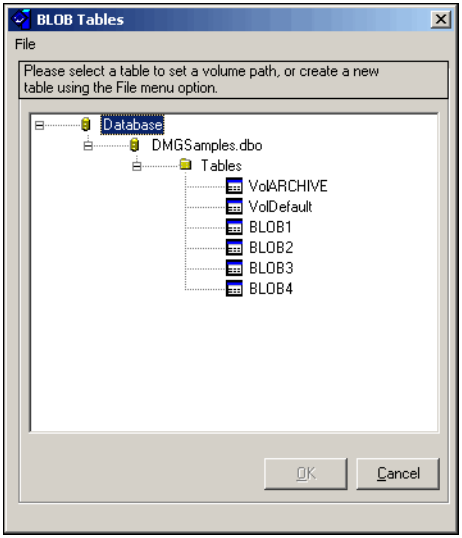
   The *Login* dialog appears.



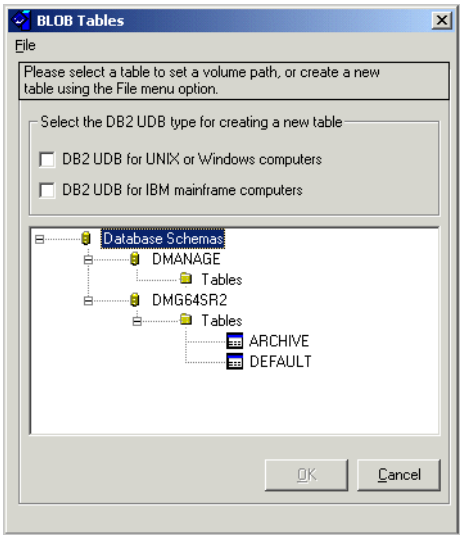
5. **Select the name of the database where the BLOB tables are located from the *Database* list.**

6 **Type your *User Name* and *Password* to connect to the database. Click *OK*.**

If you are selecting a BLOB table in a Microsoft SQL, MySQL, Sybase, or Oracle database, the following *BLOB Tables* dialog appears.



If you are selecting a BLOB Table in a DB2 database, the following *BLOB Tables* dialog appears.



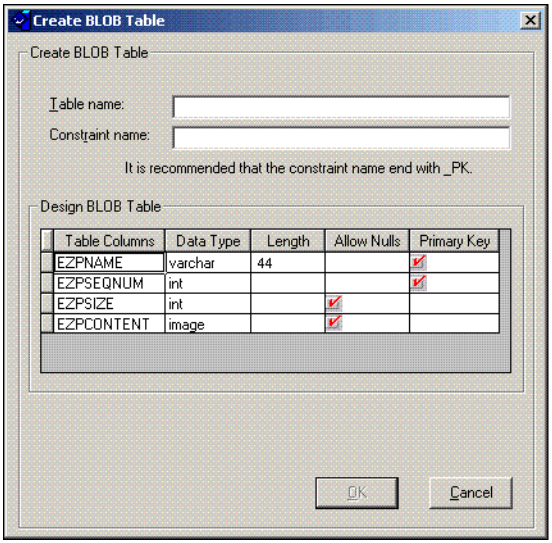
7 **Select a BLOB table and click on the *OK* button to set the volume path or create a new BLOB table as listed in the steps below.**

8 **If you are creating a BLOB table on a DB2 database, select:**

- the *DB2 UDB for UNIX, or Windows computers* checkbox if your database is running on a computer using a UNIX or Windows operating system.
- the *DB2 UDB for IBM mainframe computers* checkbox if your database is running on an IBM computer with an OS390 operating system.

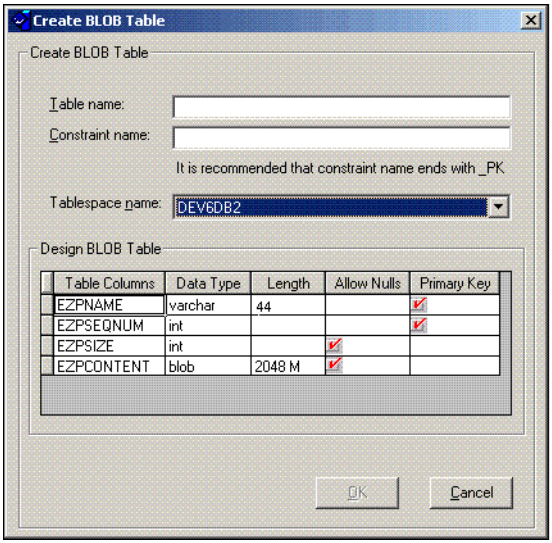
9 **On the *File* menu of the *BLOB Tables* dialog, click *Create new table* or press *[Ctrl + R]*.**

If you are creating a BLOB table in a Microsoft SQL or a Sybase database the following *Create BLOB Table* dialog appears.



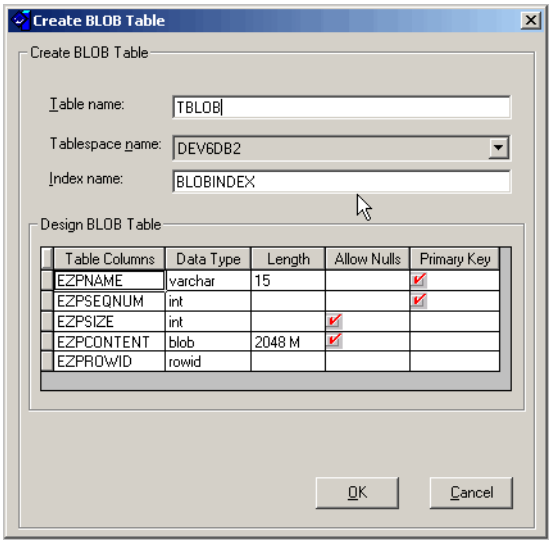
- *Table name*—Type in the name of the new BLOB table.
- *Constraint name*—Type in the name of the primary key, which should end with _PK.

If you are creating a BLOB table in an Oracle or DB2 database the following *Create BLOB Table* dialog appears.



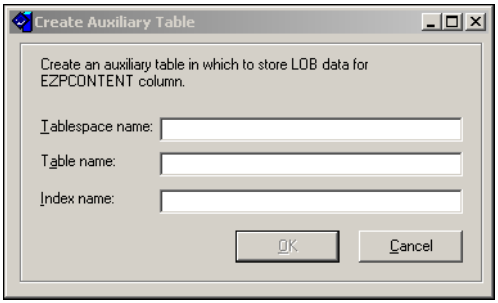
- *Table name*—Select the name of the new BLOB table from the *Table name* list.
- *Constraint name*—Type in the name of the primary key, which should end in _PK.
- *Tablespace name*—Select the name of the tablespace in which you want to create the table from the *Tablespace name* list.

If you are creating a table for a DB2 database running on an IBM OS390 operating system, the following *Create BLOB Table* dialog appears.



- *Table name*—Type in the name of the new BLOB table.
- *Tablespace name*—Type in the name of the tablespace in which you want to create the BLOB table.
- *Index name*—Type in the name of the index to the BLOB table.

If you are creating a table for a DB2 database running on an IBM OS390 operating system, the *Create Auxiliary Table* dialog appears after you click *OK* in the *Create BLOB Table* dialog.



- *Tablespace name*—Type in the name of the tablespace in which you want to create the auxiliary table.
- *Table name*—Type in the name of the auxiliary table.
- *Index name*—Type in the name of the index to the auxiliary table.

NOTE: After you fill in the boxes in a *Create Blob Table* dialog, the contents of the new BLOB table will appear.

10 **Click *OK*.**

A *Documanage Administrator* message appears indicating that a BLOB table has been sucessfully created.

Deleting a volume

WARNING: Before deleting a volume, migrate any documents that need to remain accessible to a different storage location. See “To change the location of a volume” for more information.

Deleting unnecessary volumes maintains the accuracy of Documanage volumes and keeps the list of volumes from becoming cumbersome. Use the *Storage* dialog from the Documanage Administrator menu to delete a volume.

Any documents that were stored on a deleted volume will still be accessible to Documanage users as long as the documents were migrated to a different storage location. See “Changing Volume Locations”for more information on migrating.

To delete a volume

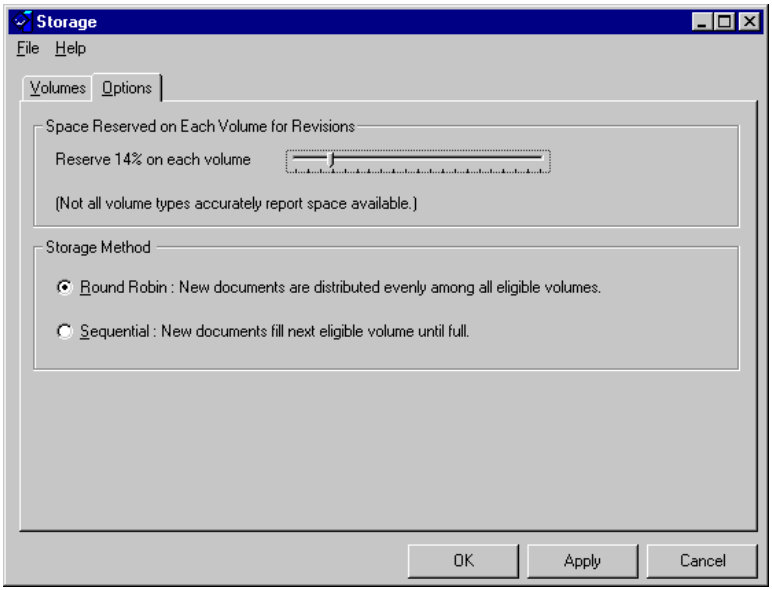
1. **With the *Storage* dialog displayed, click the volume and path you want to delete.**
2. **Select *Delete* from the File menu or press [Alt+F+D].**

   A confirmation box appears.
3. **Click *Yes*.**

   The confirmation box closes and the file is deleted.
4. **Select *Exit* from the *File* menu or press [Alt+F+X].**

   The *Storage* dialog closes and the Documanage Main Administrator dialog reappears.

Setting Storage Options



To set storage options

1. **Open the *Storage Volume* dialog and go to the *Options* tab.**
2. **Set the *Space Reserved on Each Volume for Revisions*.**

   Setting space on a volume for revisions allows you to revise a document and ensure it is stored on the same volume as the original.

3 **Select the *Storage Method* you want to use.**

NOTE: These selections have an effect only when more than one volume has been assigned to a Category.

- **Round Robin.** Documents are placed into the next consecutive volume.

  For example, if you are importing 4 documents into the system and the Category you select has 3 volumes, the first document is imported into Volume 1, the second into Volume 2, the third into Volume 3, and the fourth into Volume 1

- **Sequential.** Documents are imported into one volume until it is full, at which point documents are imported into the next volume with sufficient space.

  For example, if you are importing 100 documents and the specified Category has 2 volumes, they are all imported into Volume 1 until that volume is full, at which point the documents are imported into Volume 2.

Volume Duplication Check

Storage volume duplication occurs when two or more storage volumes point to the same secondary (or mass) storage media.

To keep data from being lost, the Documanage Server performs the Volume Duplication Check as it starts up. If more than one volume points to the same secondary storage:

- The Server writes details about the duplicate volumes (the volume names, for instance) in duplicate pairs of entries to the log. Each entry has the following form:

  ```
  VOLUME_DUPLICATION_CHECK [DUPLICATE VOLUME] causes
  server shutdown: the volume name in the OT_Volumes
  table is [DEFAULT], but the same volume is also known
  as [PRIORVOL]
  ```

  where:

  - DUPLICATE VOLUME is the volume name that is duplicated in each of the entries in the log.
  - DEFAULT is the volume name that the OT_Volumes table uses to identify the duplicated volume.
  - PRIORVOL is the alternate name for DUPLICATE VOLUME; each log entry contains a PRIORVOL name.

- The Server shuts down.

To eliminate duplicate volumes

When the Server shuts down after failing the Volume Duplication Check:

1. **Locate the log entries by searching for CVOL using the log viewer.**
2. **Read the entries to determine which of the volumes is a duplicate.**
3. **Use the *Storage* dialog to delete the duplicate. See “To delete a volume” on page 55.**

Using DMGBLOB for Storage

Normally, Documanage stores document files on the local disk or on network file system storage using the DMGNTFS storage provider plug-in. However,

Documanage can also store document files in BLOB (Binary Large Object) or Image fields within the database management system. Another storage plugin, dmgblob, allows Documanage to store document data directly on a database management system, potentially the same one that stores the data that describes the documents.

To take advantage of this feature, a new table must be added to your Documanage database or your line-of-business database for each storage volume you wish to define (Either database can be used.). Each table must have the basic set of columns shown here. Some databases may require others. The combination of the EZPName and EZPSeqnum fields form the table's Primary Key. Each file stored to the volume is written into one or more records with the same EZPName and sequential values for EZPSeqnum.

Col #	Name	Type	Size	Unique	Notes
1	EZPName	varchar	44	N	Same as file name and used in NTFS, including extension, no path
2	EZPSeqnum	int		N	Sequence number. Combination of Name and Seqnum is unique
3	EZPSize	int		N	Size of BLOB contents
4	EZPContent	BLOB		N	Actual file content, two gig max

This table can be created with SQL statements similar to the ones listed in the examples shown here.

For DB2 for Windows NT:

```
create table DMG.PO_BLOB
(EZPName varchar (44) NOT NULL,
EZPSeqnum int NOT NULL,
EZPSize int,
EZPContent BLOB (2048 M)
NOT LOGGED NOT COMPACT,
```

```
constraint bl1 primary key (EZPName,EZPSeqnum)
) in DMANAGE;
```

For DB2 MVS:

```
CREATE TABLE DMG.TBLOB
   (EZPNAME VARCHAR(44) NOT NULL
   ,EZPSEQNUM INT NOT NULL
   ,EZPSIZE INT
   ,EZPCONTENT BLOB(2048 M)
   ,EZPROWID ROWID NOT NULL GENERATED ALWAYS
   ,PRIMARY KEY(EZPNAME,EZPSEQNUM)
   ) IN DMANAGE.TSNrml
;
CREATE TYPE 2 UNIQUE INDEX DMG.BLOBINDX
   ON DMG.TBLOB
   ( EZPNAME ASC
   , EZPSEQNUM ASC
   ) CLUSTER
   BUFFERPOOL BP0
;
CREATE LOB TABLESPACE TSTBLOB IN DMANAGE
   USING STOGROUP SYSDEFLT
   BUFFERPOOL BP32K
   LOG NO
   LOCKSIZE LOB
;
CREATE AUX TABLE DMG.AUXBLOB
   IN DMANAGE.TSTBLOB
   STORES DMG.TBLOB COLUMN EZPCONTENT
;
CREATE TYPE 2 UNIQUE INDEX DMG.TBAUXIDX
   ON DMG.AUXBLOB
```

```
   BUFFERPOOL BP0
;
```

NOTE: The IBM DB2 Administrative Guide contains additional details on setting up BLOB storage on DB2 MVS.

For SQL Server:

```
create table DMG.PO_BLOB
  (EZPName varchar(44) not null,
  EZPSeqnum int not null,
  EZPSize int,
  EZPContent image,
  constraint bl1 primary key(EZPName,EZPSeqnum)
  );
```

For Oracle:

```
CREATE TABLE DMG.PO_BLOB
   (EZPNAME VARCHAR2(44) NOT NULL,
   EZPSEQNUM NUMBER NOT NULL,
   EZPSIZE NUMBER,
   EZPCONTENT BLOB,
   CONSTRAINT B11 PRIMARY KEY(EZPNAME, EZPSEQNUM)
    );
```

NOTE: You can change the “DMG” qualifier to whatever owner name you wish, and you can change the “DMANAGE” data space to the data space in which you want your BLOB to be stored.
You can also give the table any name you wish. However, you must specify the first four columns in the order shown here. Also, note that column names are case sensitive in some database installations.

DB2 MVS Table

A table is required for DB2 MVS that is very much like the one shown for other databases, However, there are additional requirements for setting up the the table:

- The Primary Key for the table must be indexed, which may have to be done manually, depending on the setting of the “CURRENT RULES REGISTER.”

- Tables and indexes are created within table-spaces/index-spaces. Those spaces define which physical volumes hold the data stored in the table and index.

- Since the BLOB table has an LOB column, two tables and table spaces are required: the original BLOB table in a normal table space and an "auxiliary table" in an LOB table space. (Each LOB column goes in a separate table and table space. In this case there is only one.) Both table spaces must be in the same database.

- Since there is an auxiliary table, an auxiliary index is also required. Also, because of the BLOB column, an additional required column of type "ROWID" is required.

There are defaults available for much of the above (database, table space, auxiliary table/index, and so on), but these offer little or no control over your system configuration.

The CURRENT RULES Register

The steps that you must execute to set up BLOB storage depend on the setting of the "CURRENT RULES" register. It controls whether certain SQL statements are executed according to the SQL Standard rules (STD) or modified rules set by the IBM DB2 MVS product (DB2). The register determines whether DB2 automatically creates the LOB table space, auxiliary table, and index on the auxiliary table for each LOB column (in this case the "BLOB" column.) If automatic creation is set up, DB2 chooses the names and characteristics of these objects.

Automatic (or implicit) creation occurs if CURRENT RULES is set to 'STD'. If you use 'STD', SQL command REPORT TABLESPACESET identifies the LOB table spaces that DB2 implicitly created. This is described in the DB2 SQL Reference.

To determine the default value of the "CURRENT RULES" register in your SQL execution environment, execute "SELECT CURRENT RULES FROM *table*," where *table* is an arbitrary table. The returned value will always be either 'DB2' or 'STD'. If nothing has set the value in your environment, the base default value is 'DB2'. Of course, you can override this setting (with the right permission) in your SQL code by using the SET CURRENT RULES = '*value*'; statement. In this statement, *value* must be either 'DB2' or 'STD'.

Limitations

BLOB fields in most database systems are usually limited to 2 gigabytes, so no file larger than 2 gigabytes can be stored in the BLOB storage subsystem.

The dmgblob plugin is currently supported only on versions 6 and greater of DB2 UDB, DB2 for OS/390 and Microsoft SQL Server.

Administering BLOB Storage

If the BLOB table is installed on a data source other than the ones known to Documanage, you must enter the data source name, type of storage provider, user ID, and password in the Configure DSNs section of the Documanage Administrator.

To install the BLOB table

1. **Go to the *Storage* section of the Administrator application.**
2. **Select *New* from the *File* menu.**

3 **Enter a unique volume name (MYBLOB for instance) in the *Volume* field.**

4 **In the *Storage Provider* field, select BLOB from the list.**

5 **In the *Path* field, enter a storage specifier.**

This specifier must be in the format: Datasource\Owner.TableName.

For example: DMANAGE\DMG.PO_BLOB

NOTE: If desired, or if required by your database configuration, additional qualifiers can be added before the table name, such as data space. Be sure that the table name exactly matches the name used in creating the table, and that the owner is the one specified in the creation script.

6 **Select *File menu* | *Save*.**

7 **Update the poffice.ini file. Use the examples shown in “Appendix B: Documanage .ini File Settings”. Look under “StorageProviders” on page 231 and “DmgSNAP storage plugin” on page 254.**

8 **To use the new volume, create a document Category and assign it to this volume as described under “Adding a new volume” on page 45.**

9 **If any Documanage Servers are running, refresh the Servers.**

Using the dmgsnap plugin for storage

Documanage provides an option for storing document data on NearStore Network Attached Storage (NAS) systems provided by Network Appliance, Inc. The NearStore product line provides WORM-like storage with customizable expiration dates. A document stored by Documanage server on NearStore cannot be deleted, even by use of a superuser account outside of Documanage, prior to the specified expiration date for the document.

A storage plugin from Docucorp, dmgsnap, interfaces Documanage to a NearStore NAS system. The Documanage storage type corresponding to the dmgsnap plugin and the NearStore product line is SNAP. The Documanage Services installer automatically installs the dmgsnap plugin in the same directory as the Documanage server executable.

NOTE: To take advantage of this feature, you must first install the NearStore product on the local area network and make it accessible (visible) to each instance of the Documanage server."Appendix G: Network Appliance Settings for Documanage" on page 309 describes how to configure the snaplock_maximum_period and snaplock_default_period values for Documanage.

To set up a NearStore Network Attached Storage system as a valid storage volume

To configure a Category and a Volume to use the SNAP storage type, proceed as follows:

1. **Go to the *Storage* section of the Administrator application.**
2. **Select *New* from the *File* menu.**
3. **Enter a unique volume name (MYNEARSTORE for instance) in the *Volume* field.**
4. **In the *Storage Provider* field, select SNAP from the list.**
5. **In the *Path* field, type the path name to the NearStore Network Attached Storage.**

   Examples:

   - For Windows — `\\mynearstore\mypath`
   - For UNIX — `/mnt/nearstore`

6 **Select *File menu* | *Save*.**

7 **Update the poffice.ini file. Use the examples shown in "Appendix B: Documanage .ini File Settings". Look under "StorageProviders" on page 231 and "DmgSNAP storage plugin" on page 254.**

8 **To use the new volume, create a document Category and assign it to this volume as described under "Adding a new volume" on page 45.**

9 **If any Documanage Servers are running, refresh the Servers.**

NOTE: In order to utilize the features of Snaplock storage for document retention, you must also configure document retention properties for at least one document Category that will use the Snaplock device for a storage volume. Checking "protect document" on the Retention configuration tab in the Categories dialog will complete the setup. Please see "Setting Retention Review Options" for details.

Using the dmgemcc plugin for storage

Using a plugin, Documanage can store documents on Centera Network Attached Storage (NAS) systems provided by EMC Corporation. Centera storage systems allow Documanage to provide WORM-like storage with customizable expiration dates. A document stored on a Centera file server cannot be deleted or altered, even by use of a superuser account outside of Documanage, before the specified expiration date for the document. A Centera file server uses Content Addressing (CA); the Centera server creates the CA, a secure alphanumeric descriptor, for each document it has in storage. CAs allow Documanage to store and retrieve documents without referencing a physical storage location, making the use of subdirectory structures unnecessary.

The storage plugin from Skywire's Documanage for Oracle, dmgemcc, interfaces Documanage to a Centera NAS system. The Documanage storage type corresponding to the dmgemcc plugin and the Centera product line is EMCC. The Documanage Services installer automatically installs the dmgemcc plugin in the same directory as the Documanage server executable.

NOTE: To take advantage of this feature, you must first install the Centera product on the local area network and make it accessible (visible) to each instance of the Documanage server. "Appendix F: EMC Centera Settings for Documanage" on page 305 describes how to set up a Centera system so that it will accept files from Documanage.

To set up an EMC Centera system as a valid storage volume

To configure a Category and a Volume to use the EMCC torage type, proceed as follows:

1. **Go to the *Storage* section of the Administrator application.**
2. **Select *New* from the *File* menu.**
3. **Enter a unique volume name (MYCENTERA for instance) in the *Volume* field.**
4. **In the *Storage Provider* field, select EMCC from the list.**
5. **In the *Path* field, type the DNS name or the IP addresses of the EMC Centera pool access node.**

   For example: `128.221.200.64`
6. **Select *File menu* | *Save*.**
7. **Update the poffice.ini file. Use the examples shown in "Appendix B: Documanage .ini File Settings". Look under "StorageProviders" on page 231 and "DmgEMCC storage plugin" on page 256.**

8 **To use the new volume, create a document Category and assign it to this volume as discussed under “Adding a new volume” on page 45.**

9 **If any Documanage Servers are running, refresh the Servers.**

Changing Volume Locations

This option provides the ability to redefine where particular document categories will be stored. If your storage needs change, move documents from one media to another. You might also do this if your current storage media is not sizeable enough.

Before a volume can be moved the new storage directory or folder must be created in File Manager or Windows Explorer. Once the storage path has been created, all new documents that you want to move can be placed in the new directory or folder. This task is accomplished through the *Storage* dialog.

To change the location of a volume

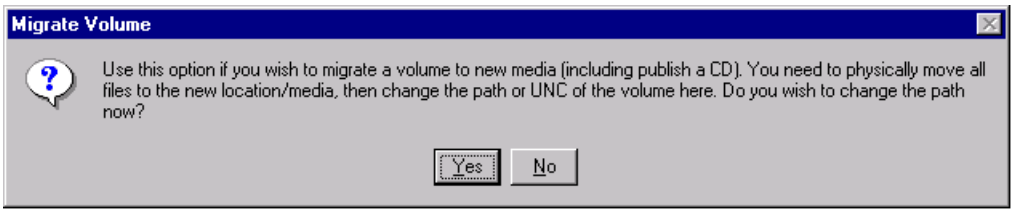
1 **In File Manager or Windows Explorer, create the new path indicating where you would like to store the documents.**

2 **Click the *Storage* option on the main Documanage Administrator dialog to display the *Storage* dialog.**

3 **Click the volume and path you want to migrate.**

4 **Select *Migrate* from the File menu or press.**

A confirmation box appears.



5 **Click *Yes*.**

The confirmation box closes and the insertion point appears in the *Path* field.

6 **Type the new path.**

7 ***Save* and *Exit*.**

Moving Documents

The Document Migration utility moves documents, based on a specified query, from their current volume to a new specified volume. Relationships and annotations are preserved. This utility can also be used to delete document versions. When deleting documents, the selected version and all previous versions are deleted. You can migrate documents through a user interface or through a command line utility.

User Interface

The document migration graphical user interface provides an interface from which you can query certain cabinets for specific documents and then migrate them to a new or different volume. You can also use the graphical user interface to delete documents. When deleting documents, the selected version and all previous versions are deleted.

With the user interface, you can:

- Create a new list
- Open an existing list
- Open an existing list with additional document attributes
- Clear a list

- Migrate a list of documents
- Delete a list of documents

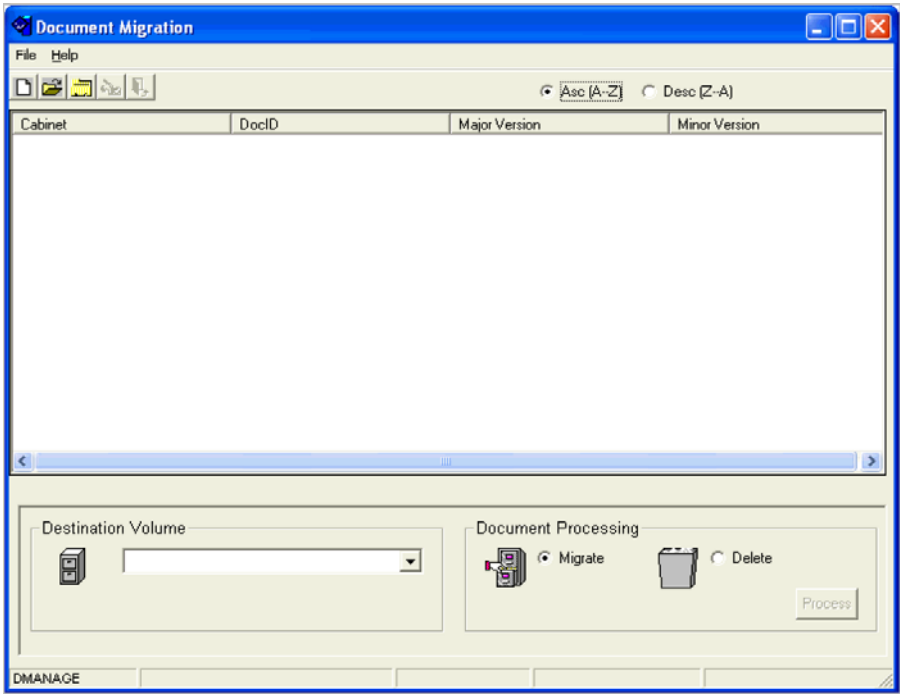
Lists contain the documents based on your query. For example, you might query the Invoices cabinet for a list of customers who have outstanding invoices. The migration tool will query the Invoices cabinet and output a list of unpaid invoices. This list can be save for later use.

Starting and Logging in to the Document Migration Utility

1. **Click the Windows Start menu.**
2. **Select Programs|Documanage|Document Migration.**

   The Document Migration login box opens.
3. **Enter your Documanage User Name, Password, and Domain in the *Document Migration Login* dialog.**
4. **Click the Router button and the enter the Router IP Address. Then click *OK*.**
5. **Click *OK*.**

The Document Migration main dialog opens.



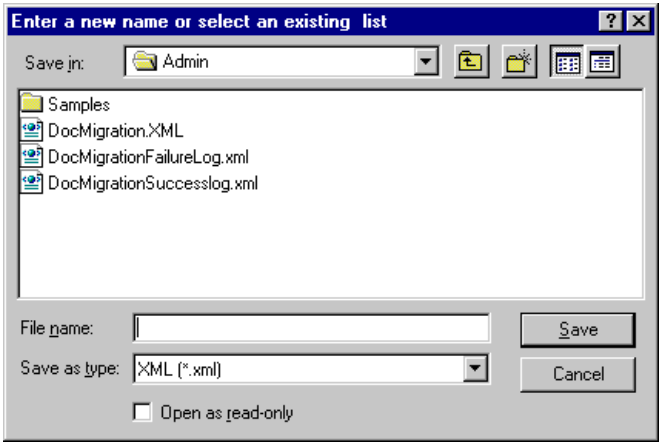
Creating a new list

Lists are created by running a query on a specified cabinet. You must create a list before you can delete or migrate documents.

1 **Open the Document Migration Utility.**

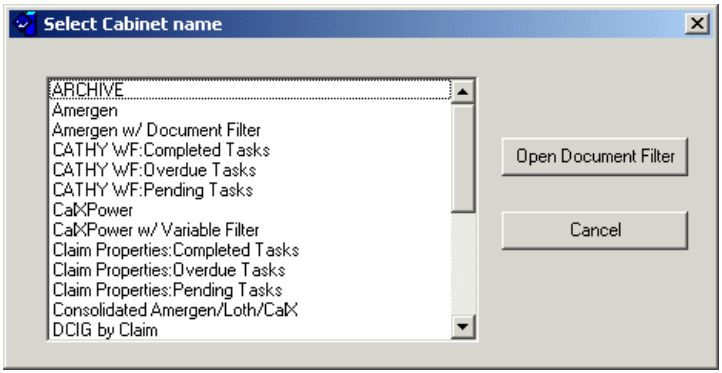
2 **Go to File|New List.**

The *Enter a new name or select an existing list* dialog opens.



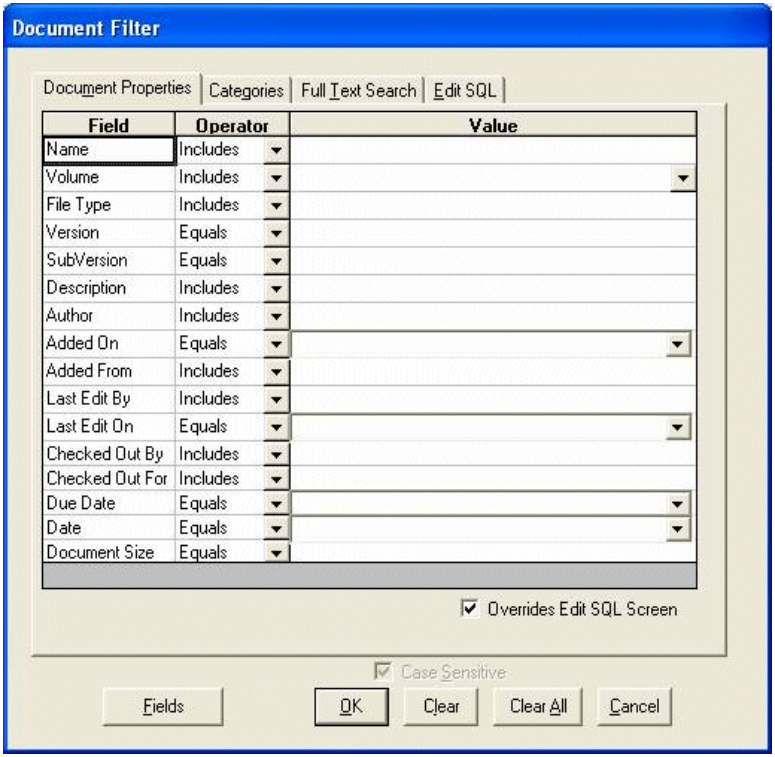
3 **Enter a name for the new list and click *Save*.**

The *Select Cabinet Name* dialog opens.



4 **Select the cabinet from which you wish to run a query then click *Open*.**

The *Document Filter* dialog opens.



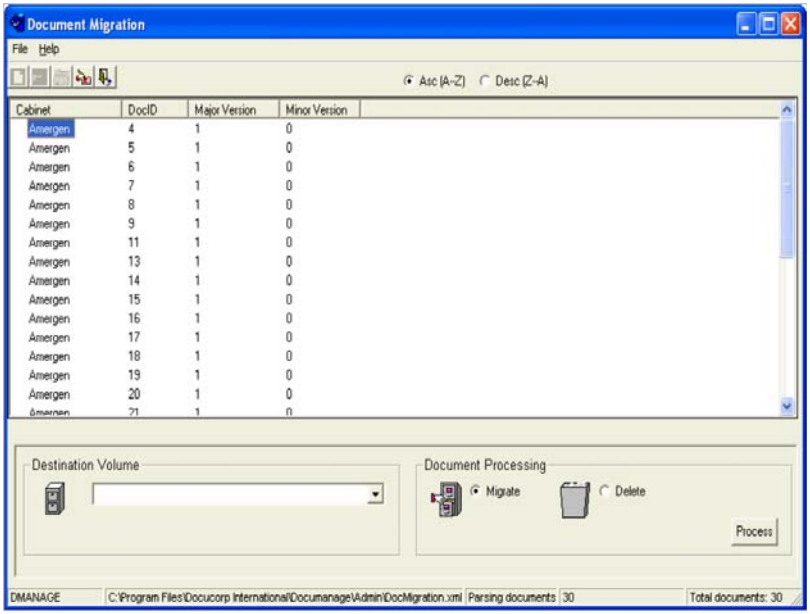
5 **Enter the search criteria you want the documents to meet on each of the tabs and click *Load Filter.***

You can search by document properties, Category, full text, or by entering an SQL clause on the advanced documents tab.

OR

You can leave the filter fields blank to receive a list of all documents in the cabinet you queried.

A list of documents displays on the *Document Migration* dialog. You can now migrate or delete the documents in this list.You can also sort the list alphabetically A-Z or Z-A by selecting Asc and Desc, respectively.



Opening an existing list

1. **Open the Document Migration Utility**

2. **Open an existing list (File|Open List|Open).**

   The *Open an existing list* dialog displays.

3. **Navigate to and select the .xml file that contains the list you want to open then click *Open*.**

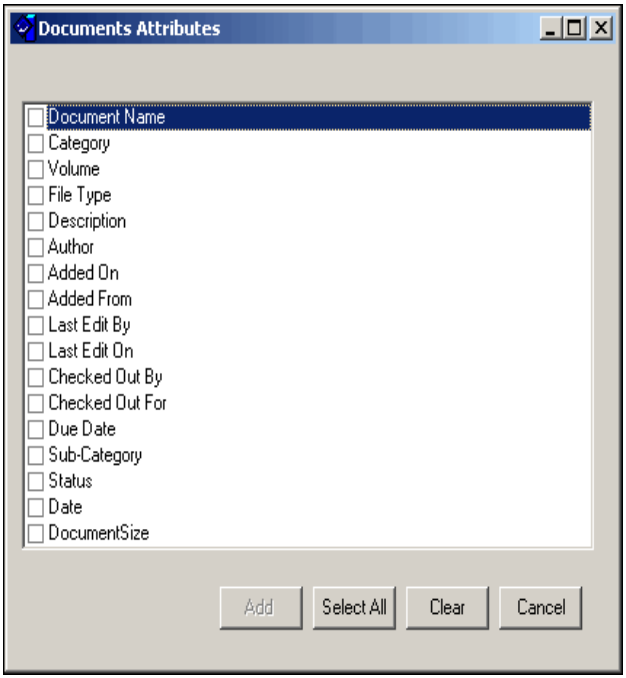
   A list of documents displays on the *Document Migration* dialog. You can now migrate or delete the documents in this list.You can also sort the list alphabetically A-Z or Z-A by selecting Asc and Desc, respectively.

Opening a list with additional document attributes

1 **Open the Document Migration Utility**

2 **Open an existing list (File|Open List|Open with additional DocAttrib).**

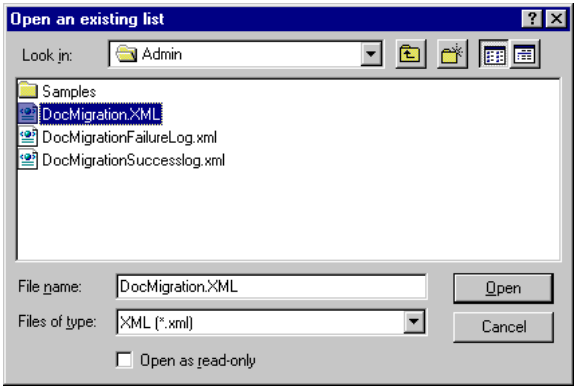
The *Document Attributes* dialog opens.



3 **Select the document attributes you want displayed on the *Document Migration* dialog and click *Add*.**

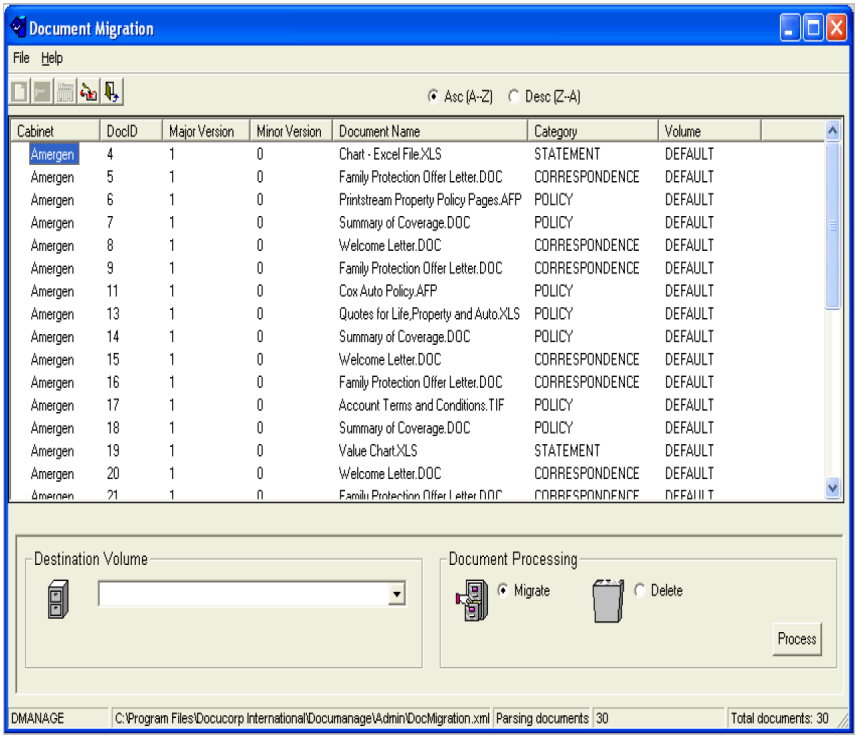
NOTE: Document Attributes are used for displaying additional document information on the Document Migration main dialog. The attributes you select are **not** saved in the XML file for future use.

The *Open* dialog appears.



4 **Navigate to and select the .xml file that contains the list you want to open then click *Open.***

A list of documents displays on the *Document Migration* dialog. You can now migrate or delete the documents in this list.You can also sort the list alphabetically A-Z or Z-A by selecting Asc and Desc, respectively.



Clearing a list

1. **Open the Document Migration Utility.**
2. **Open the list you want to clear.**
3. **Go to the *File|Clear List*.**

   The list is cleared.

Migrating documents

1. **Open the Document Migration Utility.**
2. **Create a new (File|New List), or open an existing list (File|Open List|Open).**
3. **Select the *Destination Volume* from the list.**
4. **Select the *Migrate* option from the *Documents Processing* box.**
5. **Click *Process* or select *File|Process* documents.**

   The documents are migrated to the specified volume.

Deleting a list

WARNING: When deleting documents, the selected version and all previous versions are deleted.

1. **Open the Document Migration Utility.**
2. **Create a new (File|New List), or open an existing list (File|Open List|Open).**
3. **Select *Delete* from the *Documents Processing* box.**
4. **Click *Process* or select *File|Process* Documents.**

   The selected document and all previous document versions are deleted.

NOTE: Ignore the number of failed documents in the status bar of the Document Migration dialog when the document has revised versions. Once the latest version of a document is deleted, processing other revisions in the list cause “not found” errors.

Exiting the Document Migration Utility

- **Go to File|Exit.**

Command Line Utilities

The query tool and migration tool are run from the command line. Using the query tool, you can run a query based on specified criteria using an SQL where clause on the command line. The tool then outputs a list of document specifiers in XML format. Using the migration tool, you can migrate documents to a specified volume using the data from an XML file.

Query Tool

The query tool allows you to run a query based on specified criteria using an SQL where clause on the command line. It then outputs an XML formatted list of document specifiers. The document specifiers are expressed as a set of attributes: `Cabinet(string)/DocID(number)/Major Version(number)/Minor Version(number)`.

For example, you might want to see all the invoices paid by electronic payment. You would run a query using an SQL where clause. The output would be in an XML file.

Input parameters for the query tool are:

- u: user name
- m: domain in which user resides
- w: password
- d: the document filter - a SQL where clause (optional – need if no SQL file path)
- x: the extended document filter

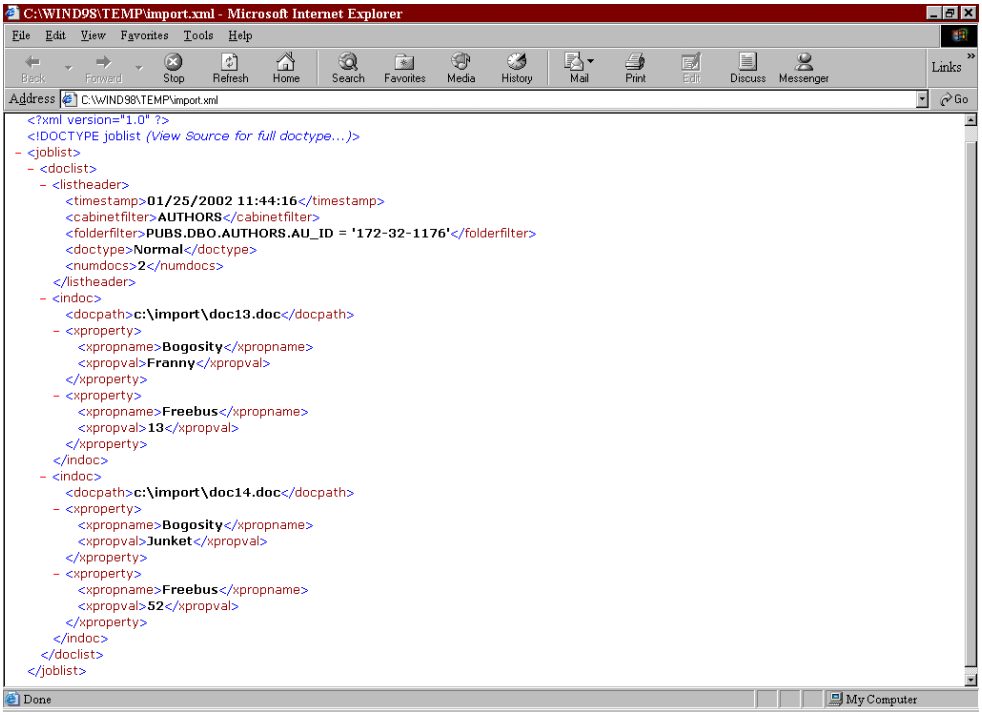
- n: a cabinet name
- t: the document Category (optional—needed for extended document properties)
- s: the path to a file containing one or more of the following: SQL filters, document Category, opcode, destination volume, and cabinet. The file is an XML file using the same DTD as the specifier file. These values are overridden by those passed in on command line (optional – need if no filters)
- l: the path to a file to be used for a log (optional - outputs to stdout)
- o: the path to a file for output of document specifiers (optional - if blank output to standard out)
- c: flag indicating that only a hit count should be returned (optional)
- b: feedback flag (optional) – tells utility to output status to command line during processing. This consists of a count of documents found and output to XML file, i.e. 1<cr/lf>2<cr/lf>…
- p: opcode (optional but recommended) – this isn't used by the tool but is output to the XML file so it is ready to process: 1=migrate; 2=delete
- v: destination volume (optional but recommended) – not used by the tool but is output to the XML file
- h, -?: outputs list of recognized commands

Sample Schema

```
<!ELEMENT joblist (doclist+)>
<!ELEMENT doclist (listheader,destina-
tion?,opcode?,(indoc*|specifier*))>
<!ELEMENT listheader (timestamp?,cabinetfilter?,docfil-
ter?,xdafilter?,folderfilter?,doctype?,numdocs?)>
<!ELEMENT timestamp (#PCDATA)>
<!ELEMENT cabinetfilter (#PCDATA)>
```

```
<!ELEMENT docfilter (#PCDATA)>
<!ELEMENT xdafilter (#PCDATA)>
<!ELEMENT folderfilter (#PCDATA)>
<!ELEMENT doctype (#PCDATA)>
<!ELEMENT numdocs (#PCDATA)>
<!ELEMENT destination (#PCDATA)>
<!ELEMENT opcode (#PCDATA)>
<!ELEMENT specifier (cabinet,docid,majorversion,minor-
version,note?)>
<!ELEMENT cabinet (#PCDATA)>
<!ELEMENT docid (#PCDATA)>
<!ELEMENT majorversion (#PCDATA)>
<!ELEMENT minorversion (#PCDATA)>
<!ELEMENT note (#PCDATA)>
<!ELEMENT indoc (docpath, xproperty*)>
<!ELEMENT docpath (#PCDATA)>
<!ELEMENT xproperty (xpropname, xpropval)>
<!ELEMENT xpropname (#PCDATA)>
<!ELEMENT xpropval (#PCDATA)>
```

Sample Import File



Error or success messages display after each command. Error messages show the error number and text. Success messages show the number of document specifiers retrieved.

Migration Tool

The migration tool uses the data from an XML file to migrate documents to a specified volume. You will usually use the XML file outputted from the query tool, but you can create or edit any XML file with the specified data. For example, your electronically paid invoice query resulted in 250 documents

from several different volumes. You can now move all the electronically paid invoice documents to an e-pay volume.

Input parameters for the migration tool are:

- u: user name
- m: domain in which user resides
- w: password
- x: the path to a file containing a formatted XML list of document specifiers, destinations and opcodes
- s: the path to a file to be used for a success log (goes to standard out)
- l: the path to a file to be used for a failure log (goes to standard out)
- p: opcode (optional) – in case it isn't already in the XML file – overrides file opcode: 1=migrate; 2=delete
- v: destination volume (optional) – in case it isn't already in the XML file – overrides file dest vol
- b: feedback flag (optional) – tells utility to output status to command line during processing. This consists of each document specifier (cabinet<sp>docid<sp>majorversion<sp>minorversion<cr/lf>)
- h, -?: outputs list of recognized commands

Error or success messages display after each command. The error message displays with text and error number. Success and failure logs are generated in the same XML format as the migration list. The success logs lists each successfully processed document specifier. The failure log lists the document that failed, its destination and opcode, and an error description in the *Note* field.

Importing Document(s)

The DMImport utility allows users to import a document(s) or folder of mixed type documents into a Documanage system. The utility operates from the command line or a batch file on the machine where Documanage is installed (the Documanage router, a server and their associated applications). The utility may also use Universal Naming Conventions (UNCs) to import documents across a network. The type of file is irrelevant to the import process, but there must be at least one defined Category and a Documanage Workstation Cabinet with each folder.

Running DMImport

On the command line, option and argument pairs must follow the DMImport executable (DMImport.exe). The pairs may occur in any order, but an option must precede each argument. Option/argument pairs may be entered at the command line, executed from a batch file, referenced through a directory, or referenced through an XML file. Referencing pairs through an XML file is the preferred way to import documents because different values can be supplied for each document.

NOTE: Document paths in the XML file must be files, not directories.

If multiple files are referenced through a directory or XML file, the attributes on the command line apply to all documents. If attributes are provided on the command line, they override any attributes in the XML file.

Syntax:

DMImport.exe <option> space <argument> space <option> space <argument>

Options:

- -c <import destination> Documanage Cabinet name
- -t <category>
- -p <path> Location of target folder or XML files.
- -f <folder filter>
- -u <User Name> This option uses login credentials listed in the DMImport.ini file—refer to “DMImport.ini File” on page 87.
- -x indicates that DMImport is taking its inputs from an XML file
- -r <string [propertyname:propertyvalue]> contains extended document properties

DMImport.ini File

If there is no .ini file, DMImport uses your Windows login credentials. If you want to use the DMImport utility with an account other than the one that your are currently logged into, you must create a DMImport.ini file and put it into the \WinNT directory. The file is formatted as follows:

```
[User]
Name=MyUser
Password=MyPassword
Domain=MyDomain

[NextUser]
Name=MyUser2
Password=MyPassword2
Domain=MyDomain2
```

If you do not use the -u option and there is an .ini file, DMImport uses the default ([User]) set of login credentials in the file. As shown here, you can add login information for other user accounts by adding entries in the files that have the same format as the first [User] entry. You can invoke these accounts by using the entry name as an argument for the -u option.

Target Folder Example

```
DMImport.exe -c Authors -t Statements -p C:\TestDocs -f
pubs.dbo.authors.au_id='213-46-8915'
```

XML File Example

Command:

```
DMImport.exe -p C:\TestDocs.xml -x
```

Minimum options/arguments required:

```
-c cabinet
-t category
-p path
-f folder filter
  -OR-
-p path (location of target folder or files)
-x flag indicates XML file
```

Document Type Definition: Sample XML File Format

NOTE: Use the INDOC element.

```
<?xml version="1.0"?>
<!DOCTYPE joblist [
<!ELEMENT joblist (doclist+)>
<!ELEMENT doclist
(listheader,destination?,opcode?,(indoc*|specifier*))>
<!ELEMENT listheader
(timestamp?,cabinetfilter?,docfilter?,xdafilter?,folderf
ilter?,doctype?,numdocs?)>
<!ELEMENT timestamp (#PCDATA)>
```

```
<!ELEMENT cabinetfilter (#PCDATA)>
<!ELEMENT docfilter (#PCDATA)>
<!ELEMENT xdafilter (#PCDATA)>
<!ELEMENT folderfilter (#PCDATA)>
<!ELEMENT doctype (#PCDATA)>
<!ELEMENT numdocs (#PCDATA)>
<!ELEMENT destination (#PCDATA)>
<!ELEMENT opcode (#PCDATA)>
<!ELEMENT specifier
(cabinet,docid,majorversion,minorversion,note?)>
<!ELEMENT cabinet (#PCDATA)>
<!ELEMENT docid (#PCDATA)>
<!ELEMENT majorversion (#PCDATA)>
<!ELEMENT minorversion (#PCDATA)>
<!ELEMENT note (#PCDATA)>
<!ELEMENT indoc (docpath, xproperty*)>
<!ELEMENT docpath (#PCDATA)>
<!ELEMENT xproperty (xpropname, xpropval)>
<!ELEMENT xpropname (#PCDATA)>
<!ELEMENT xpropval (#PCDATA)>
]>
<joblist>
   <doclist>
      <listheader>
         <timestamp>01/25/2002 11:44:16</timestamp>
         <cabinetfilter>AUTHORS</cabinetfilter>
         <folderfilter>PUBS.DBO.AUTHORS.AU_ID = '409-56-
7008'</folderfilter>
         <doctype>XDA</doctype>
         <numdocs>2</numdocs>
```

```
            </listheader>
            <indoc>
               <docpath>D:\Volume2\EZPOWER.log</docpath>
               <xproperty>
                  <xpropname>Name</xpropname>
                  <xpropval>Franny</xpropval>
               </xproperty>
               <xproperty>
                  <xpropname>Address</xpropname>
                  <xpropval>13</xpropval>
               </xproperty>
            </indoc>
            <indoc>
               <docpath>D:\Volume2\Proc.txt</docpath>
               <xproperty>
                  <xpropname>Name</xpropname>
                  <xpropval>Junket</xpropval>
               </xproperty>
               <xproperty>
                  <xpropname>Address</xpropname>
                  <xpropval>52</xpropval>
               </xproperty>
            </indoc>
         </doclist>
      </joblist>
```

Maintaining Categories

Most organizations have a high degree of structured and unstructured information within their enterprise. Structured data includes information such as name, address, telephone number, and user ID in a well-defined database. Unstructured data can include images, word-processing files, and spreadsheets located anywhere within an organizational structure. To better work within a document-management system, it is often helpful to categorize both structured and unstructured data. In Documanage, these groupings are referred to as categories.

Categories refer to a document classification, such as an invoice or purchase order. When documents are organized by Category, you can process them more easily, ensure security, allow for migration, and customize workflow. Document categories vary depending on the organization.

Related to categories are sub-categories, which are a subset of a document Category. For example, if you have a document Category called Contracts, subsets of this document Category may include Sales Contracts, Marketing Contracts, and so on. You can have as many sub-categories as necessary for your organization.

Defining document categories

The Documanage Administrator is the only person who defines document categories and provides import rights to users. Document categories can be created both when generating the Documanage system or after it has been constructed. Once document categories are defined on the Categories dialog, the user can select the appropriate Category when documents are imported into the Documanage Workstation.

For example, if the System Administrator sets up two document Categories, *Invoices* and *Contracts*, the user can select one of these types when importing documents into the Workstation.

New document Categories can be added by clicking the *New Category* button in the *Category* dialog. Document categories can be removed by clicking the *Drop Category* button in the same dialog. When you do so, that document Category will no longer be available.

The *Document Category* dialog offers further methods of working with document categories. The tabs on the dialog represent each method.

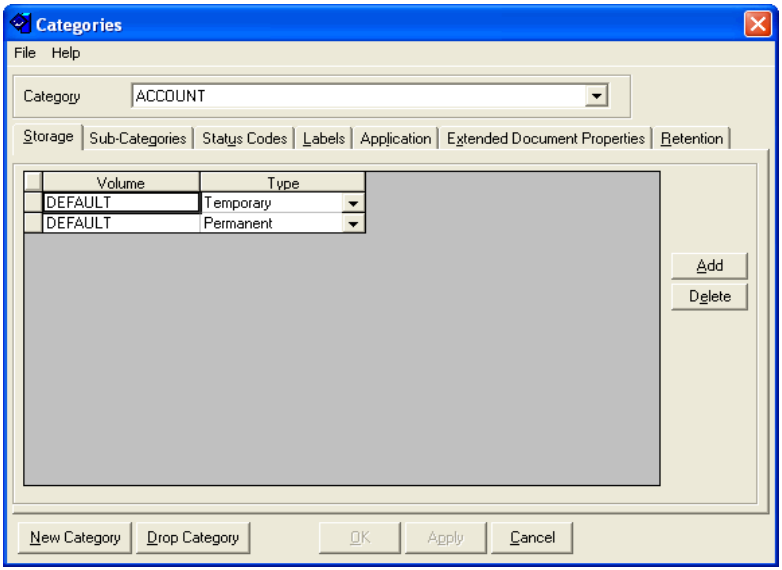
NOTE: By default, all Categories can be used when importing documents into any available Cabinets. To restrict use of certain document Categories to specific Cabinets, use the ‘Categories’ tab on the *Cabinet Definition* dialog. Please see “Creating new Cabinets” for details.

Adding and Deleting Categories

To Add a Category

1 **Click the *Categories* button on the main Documanage Administrator dialog.**

The *Categories* dialog opens.



2 **Click the *New Category* button.**

The *New Category* dialog opens.

3 **Enter a name for the document Category and click *OK*.**

Category refers to a document classification, such as an invoice or purchase order.

To Delete a Category

1. **Click the *Categories* button on the main Documanage Administrator dialog.**

   The *Categories* dialog opens.

2. **Select the document Category you want to delete from the *Category* list.**

3. **Click the *Drop Category* button and click *Yes* in the confirmation box.**

   The Category is dropped.

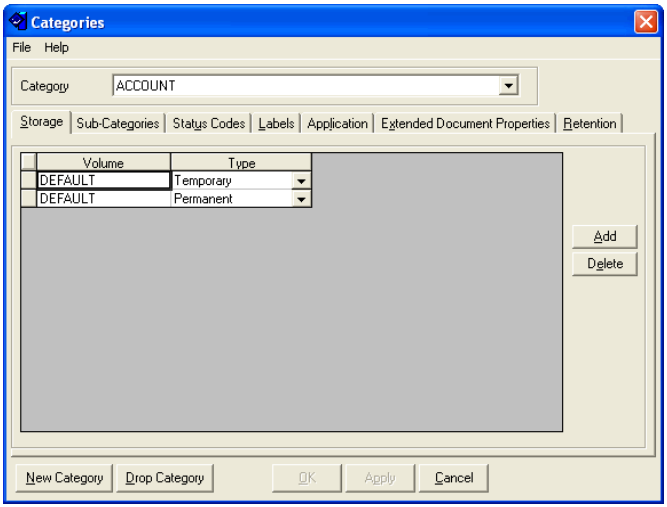
Adding and deleting storage volumes

The Storage tab in the Document Category dialog allows you to set specific storage volumes for the selected document Category. Types of storage include *Permanent*, *Temporary* and *Annotation*. Permanent storage refers to a volumes containing documents; Temporary storage refers to volumes containing checked-out documents, and Annotation refers to volumes containing document annotations.

To add a storage volume

1 **Click the *Categories* button on the main Documanage Administrator dialog.**

The *Categories* dialog appears.



2 **Select the desired document Category from the *Category* list.**

3 **Click the *Add* button.**

A list of possible volume choices appears. This list is generated from Volume information entered on the Storage dialog.

4 **Select the volume on which you wish to save the document Category from the Volume list.**

5 **Select the storage type from the *Type* list.**

- *Permanent*—*Permanent* storage contains original Documanage documents.
- *Temporary*—*Temporary* storage holds checked-out copies of Documanage documents.
- *Annotation*—*Annotation* storage holds annotation files for documents.

Each Document Category can have any number of *Temporary*, *Permanent* or *Annotation* volumes.

6 **Click *OK* to save the changes and exit the dialog, or click *Apply* to save the changes without exiting the *Categories* dialog.**

To delete a storage volume

Deleting unnecessary storage volumes eliminates outdated or invalid volumes.

1 **Go to the *Storage* tab of the *Categories* dialog.**

2 **Click the storage you want to delete.**

3 **Click the *Delete* button to the right of the *Type* column.**

A confirmation box appears.

4 **Click *Yes*.**

The storage volume is deleted.

Adding and deleting document sub-Categories

Sub-categories are a subset of a document Category. For example, if you have a document Category called *Contracts*, subsets of this Category may include

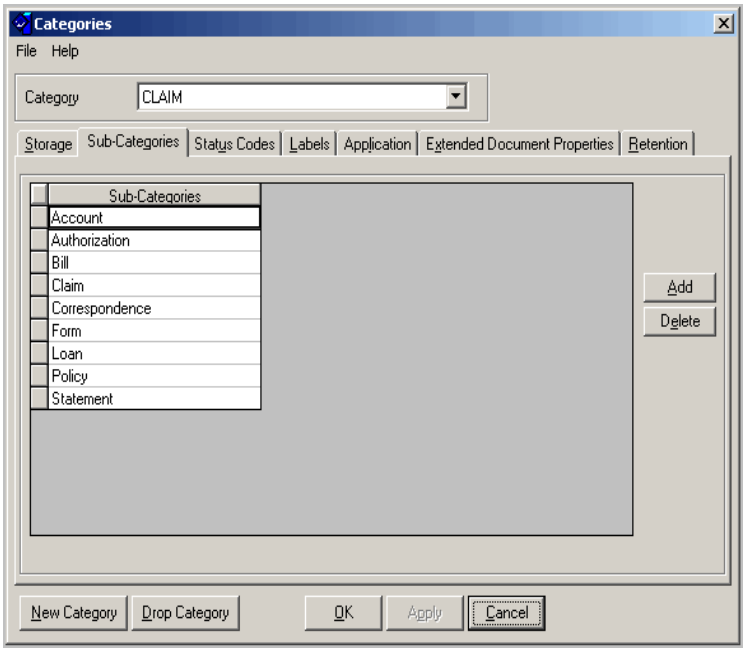
Sales Contracts or *Marketing Contracts*. You can have as many sub-categories as necessary for your organization.

Only the Documanage System Administrator can define document sub-categories on Sub-Categories tab in the *Categories* dialog.

To add a sub-category

1 **Click the *Categories* option on the main Documanage Administrator dialog.**
The ***Categories*** dialog opens.

2 **Select a *Category* and go to the *Sub-Categories* tab of the *Categories* dialog.**



3 **Click the *Add* button and enter a sub-category name.**

You have just created a sub-category within the current document Category.

4 **Click *OK* to save and close the dialog or click *Apply* to save the change without closing the dialog.**

To delete a sub-category

Deleting unnecessary sub-categories will prevent the sub-categories feature from becoming cumbersome. For example, if you had a type *Review* with sub-categories *Published*, *Reader*, and *Pending*, and you were no longer reading reviews, you could delete sub-category *Reader*.

1 **Click the *Sub-Categories* tab of the *Categories* dialog.**

2 **Select the sub-category you want to delete.**

3 **Click the *Delete* button.**

A confirmation box appears.

4 **Click *Yes*.**

The sub-category is deleted.

Adding and deleting status codes

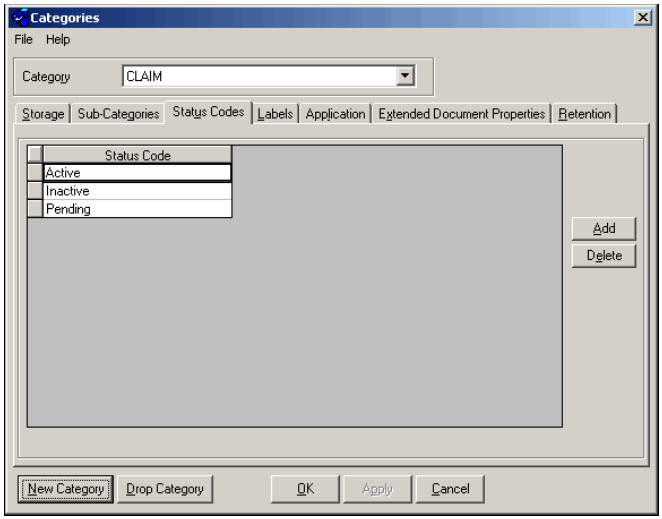
Each document brought into Documanage can be assigned a *status code*. Status codes provide a quick document workflow status. The Documanage System Administrator can create as many status codes as desired. Examples of status codes include: ***Active***, ***Expired***, ***Pending***, and ***Cancelled***. Additional status codes can be added as necessary based on organizational needs. The Documanage user can select the appropriate status code in the User module from the list created by the Documanage Administrator.

Adding a status code

1 **Select the *Categories* option on the main Documanage Administrator dialog.**

The *Categories* dialog opens.

2 **Select the desired Category and go to the *Status Codes* tab.**



3 **Click the *Add* button and enter a name for the status code.**

4 **Click *OK* to save and close the dialog or *Apply* to save changes without closing dialog.**

To delete a status code

Deleting unnecessary status codes keeps the list accurate and efficient.

1 **Click the *Status Codes* tab of the *Categories* dialog.**

2 **Select the status code you want to delete.**

3 **Click the *Delete* button to the right of the *Status Codes* column.**

A confirmation box appears.

4 **Click *Yes*.**

The status code is deleted.

Creating keywords and flags

The Labels tab on the *Categories* dialog allows you to refine a document by assigning *keyword* and *flag* attributes. Keywords are text fields that can be changed to something more specific by the Documanage Administrator. For example, for Category Contract you could have a keyword called Acme Inc. to find all contracts for Acme Inc. Keywords appear in a document's properties on the user side.

NOTE: If keywords are not specified, no keyword fields will display in the Client.

You can also set flags for a document. Flags function similarly to keywords. One difference between the two is that a Documanage Administrator can only set up to two keywords, but can set as many flags as desired under the Application tab.

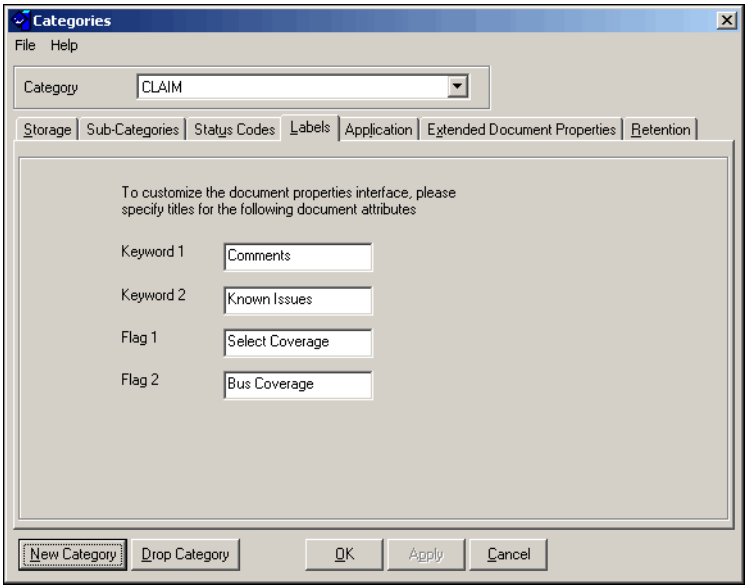
NOTE: If no flags are specified in the Application tab and none are specified in the Labels tab, no flag fields will appear in the Client.

To create keywords and flags

1 **Click the *Categories* option on the main Documanage Administrator dialog.**

The *Categories* dialog opens.

2 **Select a document category and go to the *Labels* tab of the *Categories* dialog.**



3 **Type in the new keyword or flag name in the appropriate text box.**

4 **Type in new keywords or flags for the remaining text boxes as desired.**

5 **Click *OK* to save and close the dialog or click *Apply* to save the change without closing dialog.**

Adding flags to the pick list

Once a flag is established on the Labels tab, the Administrator can create extended options for each flag. The options will display in the Client. Users

can then search for documents in the Documanage Client by these flags. These extended flags are created through the Application tab.

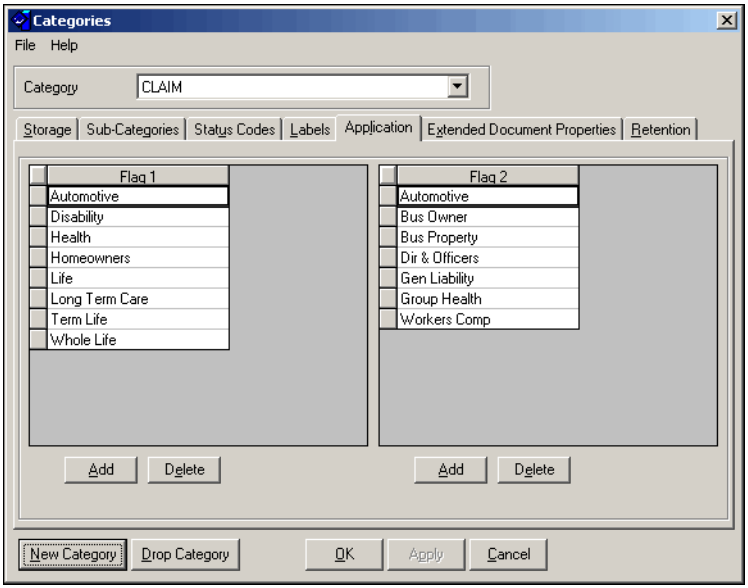
Another difference between flags and keywords is the Documanage Administrator creates a pick list for flags from which the user can choose, but only one replacement word can be made for keywords.

To add a flag to the pick list

1 **Click the *Categories* option on the main Documanage Administrator dialog.**

The *Categories* dialog appears. Ensure that flag information is set up on the Labels tab.

2 **Select a Category and go to the *Application* tab.**



3 **Click the *Add* button below the *Flag 1* column.**

4 **Type the desired Flag 1 information as desired.**

5 **Click the *Add* button below the *Flag 2* column.**

6 **Type the desired Flag 2 information if necessary.**

7 **Click *OK* to save and close the dialog or *Apply* to save changes without closing it.**

Adding Extended Document Properties

Extended document properties are user defined document properties held in the Documanage database. They are defined for each Category. You might use extended document properties to set a specific set of database properties for a Category. These new properties would behave like database fields when used in Queries, for example, but would not alter the original customer LOB database tables, since they are stored in Documanage housekeeping tables.

For example, if you are presenting an insurance bill you might want the properties to be *premium, amount due*, and *due date*.

You can use extended document properties with the built in document properties and folder criteria to search for specific documents and/or to extract critical data values from the document and make it accessible from the database without opening the document.

You define Extended document properties using the Documanage Administrator in Document Categories | Extended Doc Properties. Once entered, the database management system creates a table in the database and the document properties display in the Documanage Client on the Query By Example dialog and the Document Properties dialog.

After extended document properties are created, you need to select the *Enable Extended Doc Properties* check box on the *Cabinet* dialog for the cabinet that this document Category resides in.

Data Types

The data types that actually appear in the database vary depending on the Database Management System (DBMS) that you are using with Documanage. The Data Type list in “Step 5” names data types using different terms than some used by the DBMSs. The different terms named and described here are for the latest versions of the DBMSs used by Documanage. They could change in the future.

- DB2 (v8.2)—If you select Decimal or Numeric, DB2 enters data into the Documanage database as a “Decimal” data type, with default settings of Length 5, Precision 5 and Scale 0. As a result, DB2 can accept numbers of up to 5 digits; it truncates numbers after the decimal point. Ask your database administrator to change the Precision and Scale for the column to get the result that you want.

  If you select Float, DB2 enters data into the database as a “Double” data type, with a default setting of Length 8. The data that DB2 maps to the Real, Smallint, Int, Double, and Float data types is subject to the limits of DB2. For example, the acceptable range for a Smallint is between -32766 to 32767; DB2 rounds numbers after the sixth decimal position for Float and Double data types.

- SQL (v8.00.760)—If you select Decimal or Numeric, SQL enters numbers into the Documanage database with default settings of Precision 18 and Scale 0. As a result, SQL truncates numbers after the decimal point. Ask your database administrator to change the Precision and Scale settings for the column to get the result that you want.

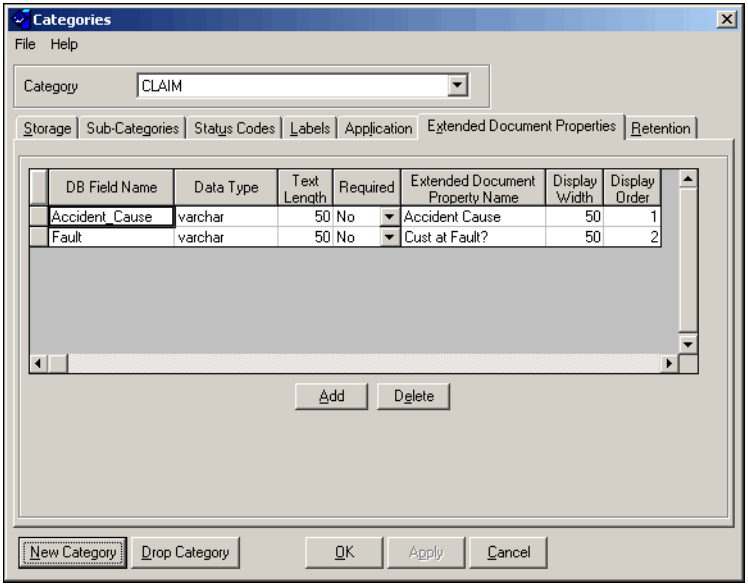
  Data that SQL maps to the Real, Smallint, Int, Money, Smallmoney, and Float types is limited by the SQL standard.

- Oracle (v10.g)—If you select Decimal, Oracle enters numbers into the database as "Number" data types with the Precision and Scale default settings blank. As a result, Oracle truncates numbers after the decimal point. Ask your database administrator to change the Precision and Scale settings for the column to get the result that you want.

  If you select Double Precision, Oracle enters numbers into the database as "Float" data types with default settings of Precision 126 and Scale blank.

1. **Click the *Categories* button on the main Documanage Administrator dialog.**

   The *Categories* dialog opens.



2. **Select the Category to which you want to add Extended Document Properties from the *Category* list.**

3. **Go to the *Extended Doc Properties* tab and click *Add*.**

4 **Enter the *DB Field Name*.**

This is the name that will be created in the database.

5 **Select the *Data Type* from the list.**

This is the type of data in the database.

NOTE: The data types that actually appear in the database vary depending on the Database Management System that you are using with Documanage. In the tab, a few data types are not named using the same terms. Refer to “Data Types” on page 104.

6 **Enter the *Length.***

This is the length of the field.

7 **Select whether or not it is required in the *Required* list.**

Selecting *Yes* requires the user to enter information for this in the Client.

8 **Enter the *Extended Document Attribute Name.***

This is the name that will display in the Client.

9 **Enter the *Display Width* and the *Display Order.***

This is the width of the field and the order in which it will display in the client.

10 **Click *Apply* to apply your changes or *OK* to apply the changes and exit the form.**

Selecting Document Retention Options

Retention Management keeps users from altering or deleting documents by putting automatic controls on how long documents will be retained and when they will be deleted. Organizations use Retention Management to implement

policies for retaining document versions. Such policies can require deleting document versions when they are no longer required.

Documanage Retention Management allows users to store document versions on third-party file servers which provide write-once read-many (WORM) -like storage that is time-limited. These servers provide secure protection for document versions, making it almost impossible to delete documents, even with an account that has administrative privileges.

In addition, Documanage can manage the retention of files on NTFS, VLAM and BLOB storage volumes, but without the security features provided by WORM-like third-party file servers.

Retention Reviews

A Retention tab in the Category dialog in the Documanage Administrator application allows an administrator to set retention options for each document Category. These options cannot be changed by Workstation users. Retention Management options specify how long the Documanage Server will keep a document version under retention management, and how the retention status of the document version will be evaluated at the end of the retention period. Document versions in a Category can be retained for a set time interval after they are imported into the system, until a specific fixed date, or forever. Retention options are applied to a document version when it is imported into Documanage and at its review time.

A review time occurs when (and if) a document version's retention period ends. At review time, the Documanage Server automatically reviews the options that currently apply to the Category for the document version and then applies the current retention options for the document's Category. Review time rules can cause a document version to be:

- retained forever, or until a later date;

- released from retention management to a normal (un-retained) status;
- or deleted

NOTE: Document versions that do not have retention options applied to them when they are imported into Documanage will not be affected by retention management, since they will never be subject to retention review. These versions include those imported into Documanage before a version of Documanage Server with Retention Management is installed and those imported using a Category that did not have retention management enabled.

Retention Review Rules

Document versions that are not being retained forever are subject to review. The following review time rules can cause a document version to be retained forever, or until a later date.

- If the current retention option for the Category is to retain document versions for a longer duration, or for a later fixed date, the Documanage Server retains the document version until the later time. This can occur when the review time established by the current retention review option differs from the one specified when the document was imported in to the system.

  For example, if the retention interval has increased, then Documanage calculates a new retention review date by adding the new retention interval to the date on which the document version was added to the system.

- If an earlier version of the document version under review has a later review date, the Documanage Server assigns the document version under review the later review date.

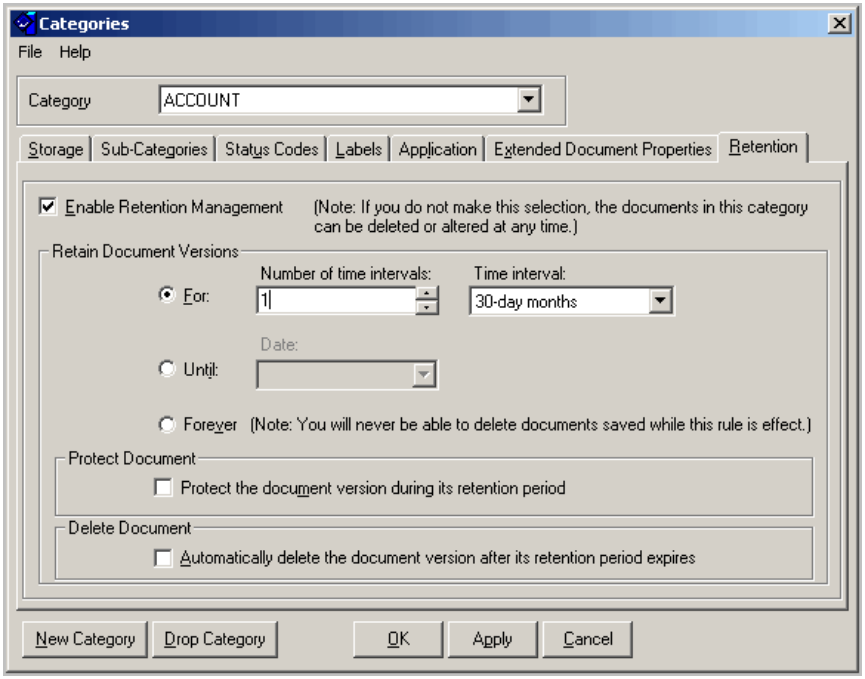
  This rule applies if the document version under review is not the oldest version. An earlier version might have a later review date if, for instance, a prior version was stored in a different Category, or if the prior version was imported into the system before the retention time for the Category was reduced, and the current document version was imported into the system after the retention time was reduced.

- If the current retention management option for the Category is to retain document versions forever, the Documanage Server retains the document version forever, with no later retention review time.

- If more than one of these rules applies, the Documanage Server applies the latest review date to the document version under review. For instance, if a prior version has a review time one week later, but the Category's current retention option extends the review time for six months, then Documanage reviews the document in six months.

Setting Retention Review Options

As shown here, the Documanage Administrator uses the Retention tab in the Categories dialog to enable Retention Management and assign Retention Review options to each document Category.



- Enable Retention Management—select Enable Retention Management to put documents subsequently imported into the Category under Retention Management. If you do not make this selection, users will be able to delete or alter document versions imported into the Category at any time, and document versions will not have retention reviews.

The Retention Management options include how long document versions are to be retained, whether they are deleted after a retention review, and whether

they are protected from deletion or modification by Documanage while under retention:

- Retain Document Versions:
  - *For*—protect document versions for a set interval after they are imported into Documanage. You can set the interval in days, in weeks, in 30 day months, or in 365.25 day years. You select the number of intervals using the *Number of time intervals* box, and you select what each interval represents using the *Time interval* box. If you select another time unit from the *Time interval is* box, the previous number of intervals is converted into the present number of intervals.

    Documanage adds the interval to the date that the document version was imported to calculate its next review time. The time interval can be up to 200 years from the date on which it is set or up to December 31, 9998, whichever is earlier.
  - *Until*—protect all of the documents imported into the Category until a common review date. You set the review date by entering it directly into the date box in mm/dd/yyyy format, or by selecting from the calendar that drops down from the box. Documanage reviews the retention status of all of the documents in the Category on this date, no matter when they were imported into the system.

    The retention review date may be up to 200 years from the date on which it is set, or up to December 31, 9998, whichever is earlier.
  - *Forever*—protect document versions indefinitely. Do not schedule reviews.
- Two selections modify how these options affect document versions:
  - *Protect Document: Protect the document version during the retention period.* —Keep protected documents from being deleted or altered by

Documanage during retention. If available, protection is enforced using the security features of third-party file servers, which make it impossible to delete documents, even with an account that has administrative privileges. Making this selection protects all retained document versions from deletion and modification within Documanage, but external third-party enforcement can only be applied to versions that are directed to one of those servers; document versions that are directed to other types of servers or volumes will not receive secure protection outside of Documanage.

- *Delete Document: Automatically delete the document version after its retention period expires*—Delete a document version after the next retention review. Documanage re-evaluates the document's retention status if a future retention time does not apply. If a future retention time does not apply and the checkbox is not selected, the document version will no longer be under retention management; it will no longer be protected and it will not have a scheduled retention review date.

CAUTION: Make sure that you want the document versions in this Category deleted before you make this selection. Selecting this option may result in the deletion of large numbers of document versions when review dates occur.

Maintaining system profile options

The *System Profile* dialog allows you to set Default Storage Locations for various documents. It also allows you to set transaction logging and timeout options.

The Documanage Administrator can define default storage options in Documanage so that whenever a personal document enters the system, a default document Category is displayed. Similarly, when annotation files or recovery files are created, a default document Category would be established. Naturally, the Documanage Administrator can change these defaults when defining new document categories.

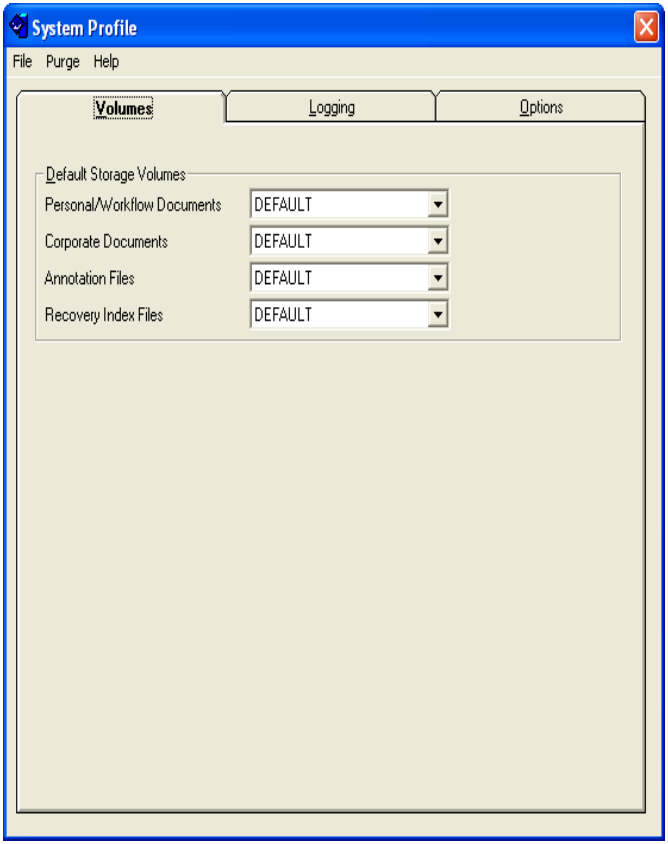
The Documanage System Administrator can use transaction logging to monitor or track system usage and maintain Documanage at optimum efficiency. Transaction logging options include adding an annotation layer, deleting an annotation layer, copying a document, and moving a document. The Documanage Administrator logs these operations in Documanage for later use.

The transaction log that is created may be of assistance in identifying bottlenecks in the Documanage system as well as determining whether existing security measures are sufficient in protecting your information. The Documanage Administrator can then generate transaction reports using third-party reporting software such as Crystal Reports and other well-known packages. With reporting software, the Documanage Administrator can produce detailed usage summaries of Documanage functions, such as the number of users who edited a document, the number of users who checked out a document, and so on.

To change system profile options

1 **Click the *System* button from the main Documanage Administrator dialog.**

The *System Profile* dialog opens.

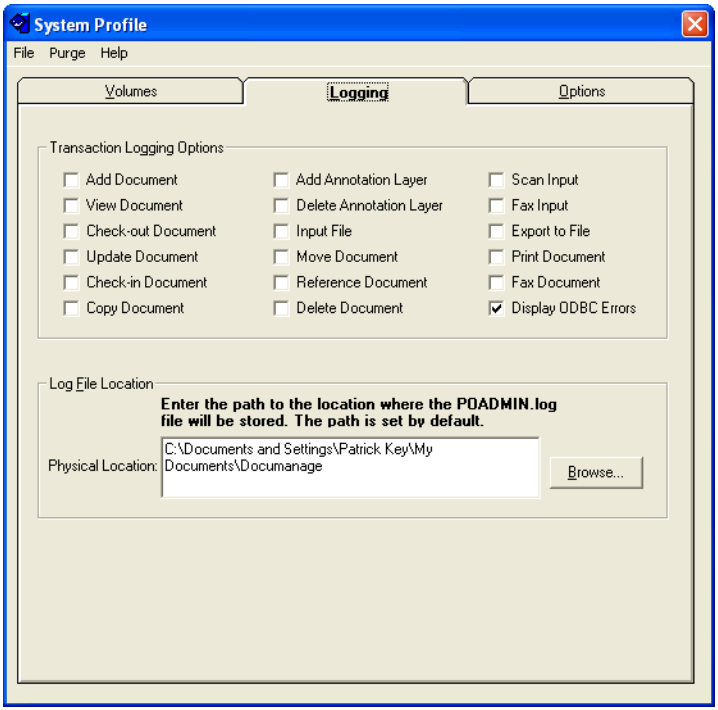


1 **To change default document storage volumes:**

- Select the appropriate drop-down list in the *Default Storage Volumes* area at the top of the *System Profile* dialog.
- Select the appropriate document Category.

2 **To set transaction logging options:**

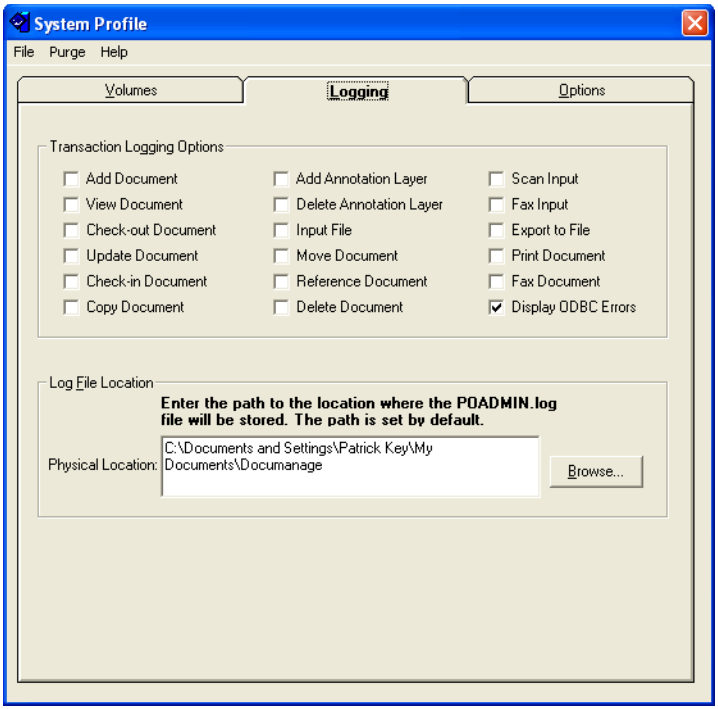
- Select the transaction you want the system to track. The Documanage Administrator will now log this option. The selected transactions can be analyzed through third-party software such as Crystal Reports.



3 **To set the *Log File Location*:**

- By default, the poadmin.log file is created in the `\Documents and Settings\ $User\MY documents\Documanage` directory. The location of the poadmin.log file is written into the pooffice.ini file.

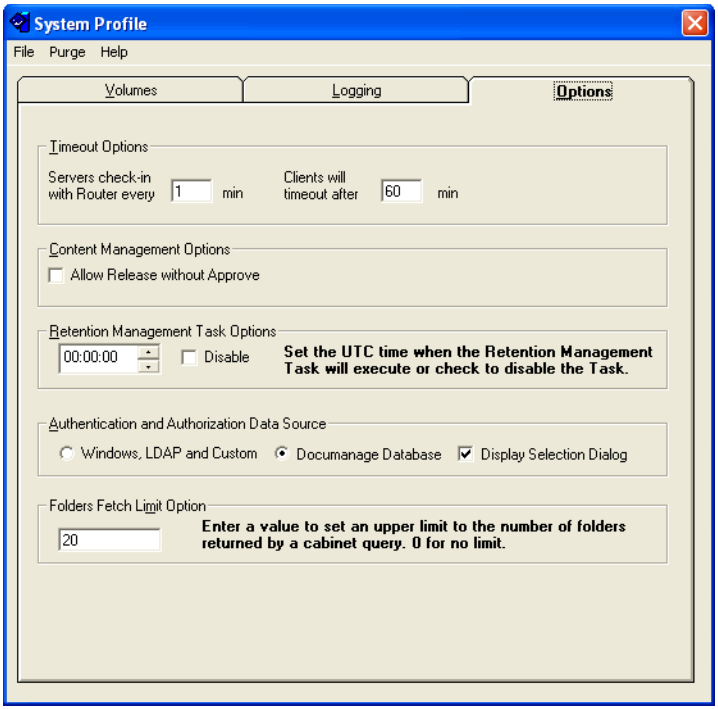
- You can browse to an alternate location for the log file directory using the *Browse* button on the *System Profile* dialog.



4 **To set timeout options:**

- In the Servers check-in with Router every field, type a value for how often (in minutes) you want the Documanage Servers to check-in with the Documanage Router. Keep this value as low as possible.

- In the *Clients will timeout after* field, type a value for when (in minutes) you want inactive clients to timeout.



5 **To set content management options:**

- Select *Allow Release without Approve* to allow documents to be released without first being approved.This is primarily intended for use with Oracle's Docuflex product, but is available for general use when Docuflex is not part of your solution.

6 **To set retention management task options:**

- Set the UTC (Coordinated Universal Time) value when the server should execute the retention management task. This task evaluates

and applies all of the defined retention management rules. Select the Disable checkbox to disable the task.

7 **To set the Authentication and Authorization Data Source:**

- Select *Windows, LDAP and Custom*
- Select *Documanage Database*
- Select *Display Selection Dialog*

8 **To set the Folders Fetch option:**

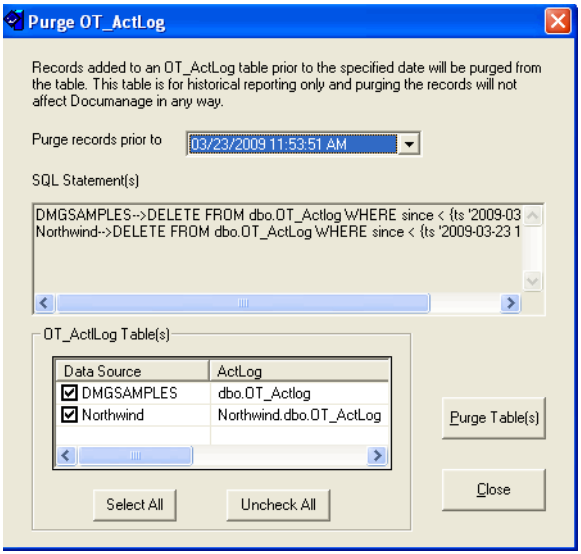
- Enter a value to limit the maximum number of folders that will be returned when a cabinet is opened. This option can be used to enhance performance on systems with a very large number of folder records; however, it will prevent users from accessing any folders over the limit. Enter zero for no limit (the default).

9 ***Save* and *Exit*.**

Purging the Activity Log Table

Documanage maintains an OT_ActLog table for each of the databases that have been Powermapped. Various events are recorded in this table and the records can be used for reporting and auditing. This table can grow quite large and there is now an option available to purge the table of older records.

1 **Select the Purge menu on the System Profile dialog and select the OT_ActLog Table… option. The Purge OT_ActLog dialog is presented.**



2 **Select the "Purge records prior to" field and a calendar will be presented for you to pick a date.** Records timestamped prior to that date will be purged from the table**.**

3 **A list of the powermapped databases is presented (commonly this will only be one database). Check each database that should have its OT_ActLog table purged.**

4 **The dialog will present the SQL statement(s) that will be executed against the table(s). This is for informational purposes only. Select the "Purge Table(s)" button to purge the records from the table(s). The statements will be executed and an informational message will be posted.**

5 **Click "Close" to close the dialog and return to the System Profiles dialog.**

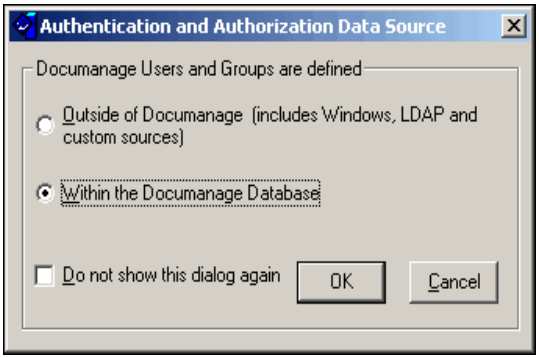
Maintaining the Users list

Documanage can use three user authentication mechanisms. It can use Windows NT security, a user/group management scheme that uses the Documanage database, or a Lightweight Directory Access Protocol.

When using Windows security, Documanage inherits users and groups defined in Windows. Maintenance functions, such as adding new users, updating passwords, and changing group composition, can all be performed from Windows. Users login using their Windows password. Users in multiple domains can share a Documanage installation.

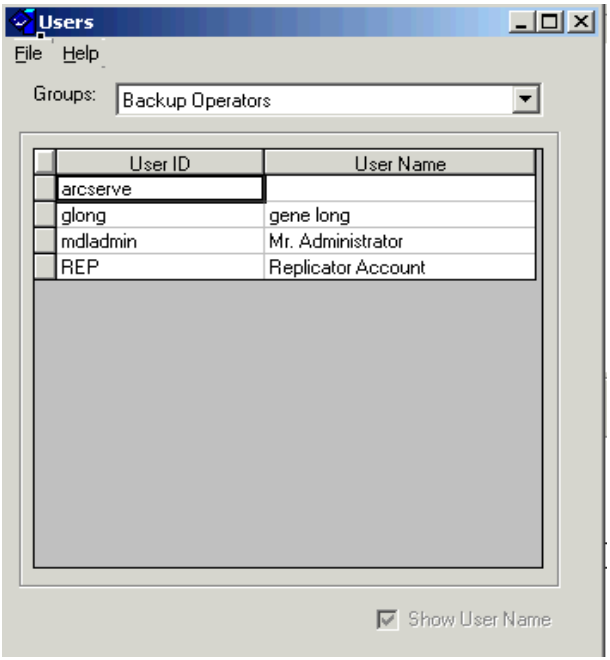
Note that only authentication is performed by Windows; authorization (user rights and permissions on various cabinets, folders and documents) are still defined and enforced within Documanage.

Clicking on *Users* in the main Administrator dialog displays the *Authentication and Authorization Data Source* dialog as shown here. This dialog allows you to view the users in either the Windows NT domain or in the Documanage database. Select the "Do Not Show This Dialog Again" check box if you wish to go directly to the user model you have selected in the future (you can also change the access model from the *System Profile* dialog).



NT Users

If you are a Documanage Administrator who wishes to see a list of Documanage system users sorted by group, click *Outside of Documanage* to view a list of current Documanage users on the *Users* dialog. You will see each user's *User ID* and *Full Name*.



You cannot add users through this dialog since it uses Windows NT security. To add, modify or delete users when using NT Security, an NT administrator must use the tools supplied by Microsoft for administering Users and Groups for the domain or the computer as appropriate.

Maintaining the NT Users list

1 **Click the Users option on the main Documanage Administrator dialog.**

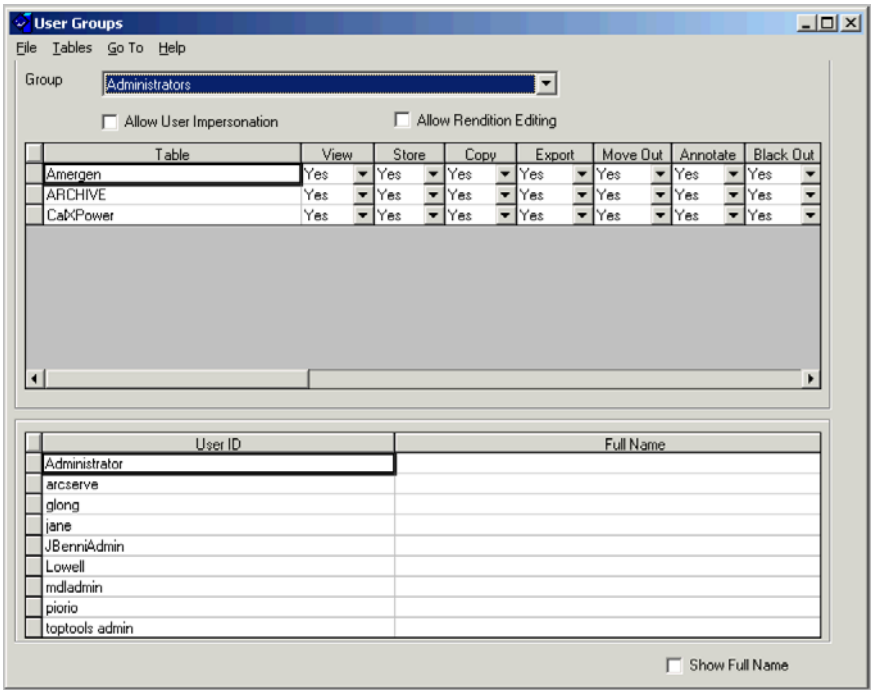
The *Authentication and Authorization Data Source* dialog appears.

2 **Click the *Outside of Documanage* radio button and click OK.**

The Users dialog displays, listing current Documanage users.

3 **Select the group of users you want to view.**

The users in the group you selected are listed.



Documanage Database Users

The *Authentication and Authorization Data Source* dialog allows you to select security *Outside of Documanage*, or *Within the Documanage Database*. The *Within the Documanage Database* option will display the *Database Users* dialog, where you can view and edit users in the Documanage database. This dialog displays the *User ID*, and the user's *Last*

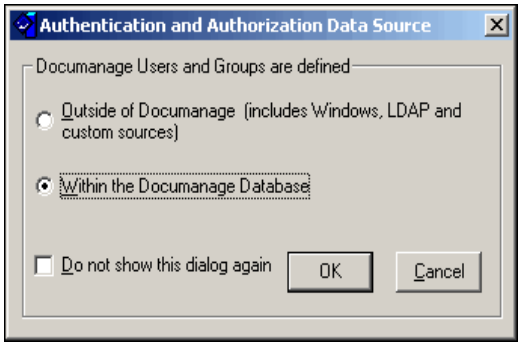
Name and *First Name*. Click *New User*, *Delete*, or *Modify* on the bottom of the dialog, to create a new user, delete a user, or modify an existing user.

NOTE: Please also see the POFFICE.INI discussion in the Appendix to change authorization settings for Database security.

Maintaining the Database Users list

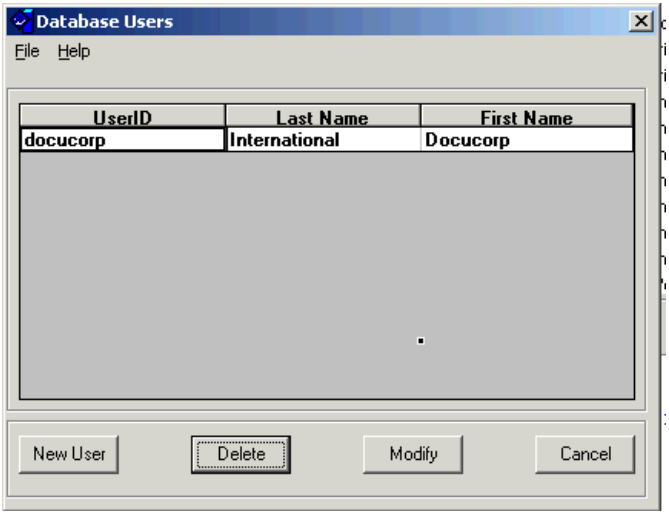
1 **Click the Users option on the main Documanage Administrator dialog.**

The *Authentication and Authorization Data Source* dialog appears.



2 **Select *Within the Documanage Database* and click OK.**

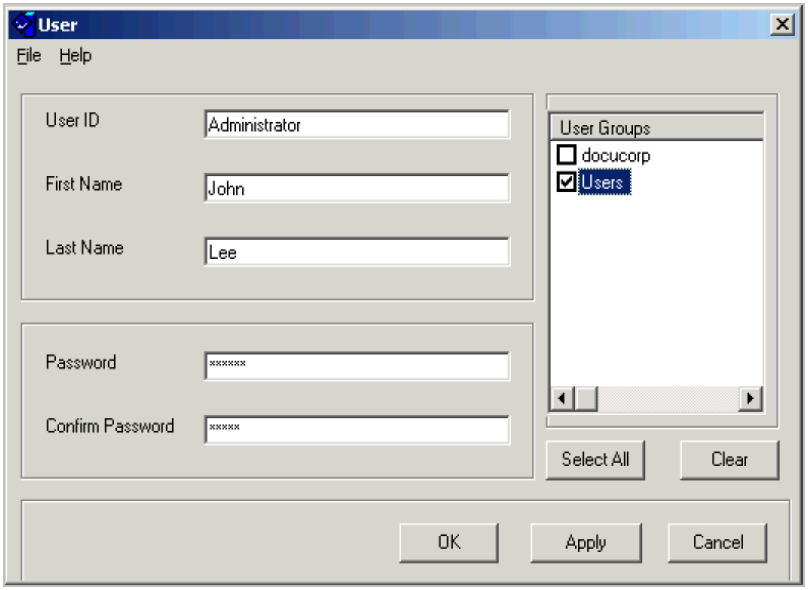
The Database Users dialog appears, listing current Documanage users.



To add a user:

1 **In the *Database Users* dialog, click *New User*.**

The *User* dialog appears.



2 **Enter the *User ID*, *First Name*, *Last Name* and *Password*.**

NOTE: Restricted characters for User and Group names within the Documanage Database are / \ | , ? * “.

3 **Select the checkboxes in the *Groups* list for any groups the user should be a member of.**

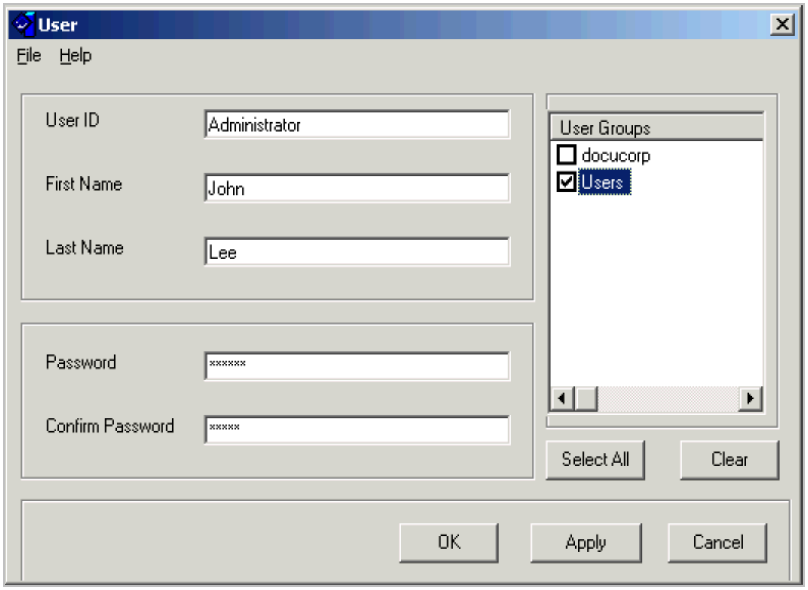
NOTE: In the *Groups* list, the *Users Group* entry is selected when the *User* dialog appears, since it is the default group used for Documanage server logon rights. To change this group, edit the `LoginGroup=Name` in the `Router` entry section of the POFFICE.INI file.

4 **Click *OK*.**

To modify an existing user:

1 **Select the user entry from the *Database Users* window and click the *Modify* button.**

The *User* dialog appears.



2 **Change the user information. Note that User ID *cannot* be modified. Click *OK*.**

To delete a user:

1 **Select the user from the *Database Users* window and click the *Delete* button.**
A dialog appears asking you to confirm that you want to delete the user.

2 **Click *OK*.**

NOTE: See also the POFFICE.INI discussion in the Appendix to change authorization settings for LDAP security.

Managing an LDAP Users List

The Lightweight Directory Access Protocol, or LDAP, queries and modifies directory services running over a TCP/IP network. The LDAP plugin supplied with Documanage interfaces a Documanage server to an LDAP server. This allows the Documanage server to use data from an LDAP server to authenticate and authorize Documanage users. Settings stored in the LDAPAuthConfig section of the poffice.ini file configure the plugin.

To utilize LDAP security, select the *Outside of Documanage* option in the *Authentication and Authorization Data Source* dialog. For further LDAP user configuration options, please see the Appendix on LDAP use with Documanage.

NOTE: See also the POFFICE.INI discussion in the Appendix to change authorization settings for LDAP security.

JXplorer

You can use an open-source LDAP-compliant browser like JXplorer to manage an LDAP directory. JXExplorer can read and search an LDAP directory, or any X500 directory with an LDAP interface. JXplorer performs the operations required to browse and modify an LDAP directory. It is also highly configurable—the user can customise it in a number of ways, ranging from the configuration of HTML viewing templates to the creation of Java plugins.

To download JXplorer, go to: <http://www.jxplorer.org/>

Maintaining user security

Security is crucial for any enterprise. Documanage provides two dialogs from which you can assign security attributes to Documanage users: the User Authorities dialog and User Groups dialog.

NOTE: You can easily jump between the User Authorities, User Groups, and Business Data Tables dialogs by selecting the corresponding option from the Go To menu.

Using the User Authorities and/or User Groups dialogs, the Documanage Administrator can determine the exact Documanage functions that a user group may perform by table. Security attributes may be “mixed and matched” as needed to implement the desired security level. These attributes may be changed at any time as your organization grows or changes, including adding groups or deleting a group’s access to a particular table.

When using the database access model, the User Groups dialog can also be used to assign users to groups. Groups can also contain other groups, but only nested one-deep.

Security attributes may be set either to Yes or No. An SQL selection is also available for more specific SQL Where clauses. A complete list of security attributes and their definitions are provided below.

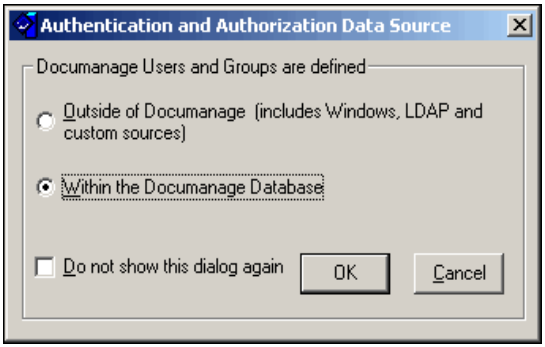
Feature	Function
View	Allows a user to view a document
Store	Allows a user to add a document as an original, checkout copy or shortcut
Copy	Allows a user to make a copy of the document
Export	Allows a user to export a copy of the document
Move Out	Allows a user to move a document from one Documanage location to another
Annotate	Allows a user to add an annotation layer
Black Out	Allows a user to add a blackout layer
Edit Contents	Allows a user to update the contents of a document
Edit Profile	Allows a user to update a document's properties
Check-In without History	Allows a user to check-in a document without creating a new version
Create Major New Version	Allows a user to create a new major version of a document (By default, users can only create a new minor version.)
See History	Allows a user to view a document's history. You must set this to Yes if you want users to create renditions, since their rendition sets may contain historical document versions.
Delete	Allows a user to delete a document

Feature	Function
Folder Delete	Allows a user to permanently delete a folder, which deletes the underlying data record
Folder Insert	Allows a user to create a new folder
Folder Update	Allows a user to edit a folder's properties
Folder View	Allows a user to view a folder
Admin	Allows a user to administer a table. This attribute allows a user to assign or revoke any rights (including Administrator rights) to/from other users

To assign security via the User Groups dialog:

1 **Click the *User Groups* option on the main Documanage Administrator dialog.**

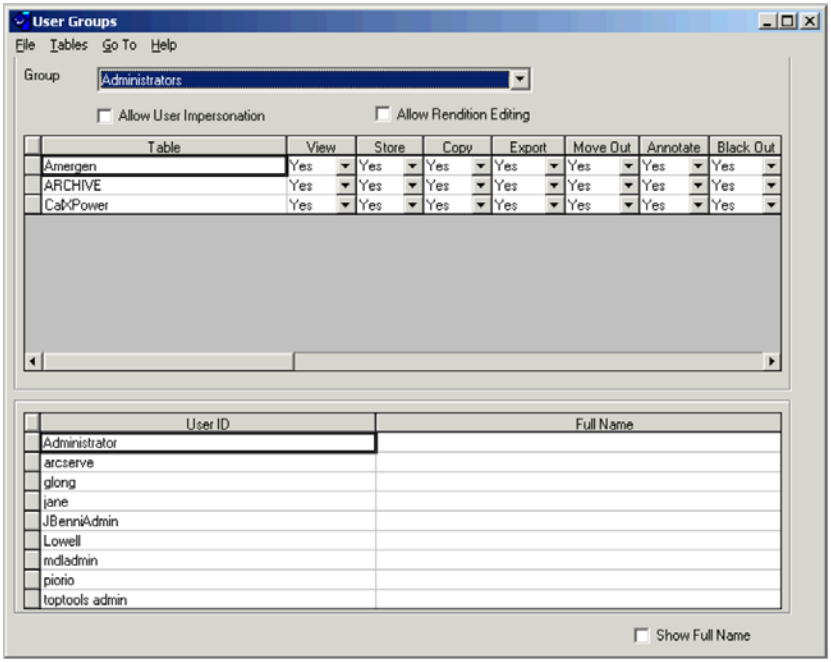
The ***Authentication and Authorization Data Source*** dialog appears.



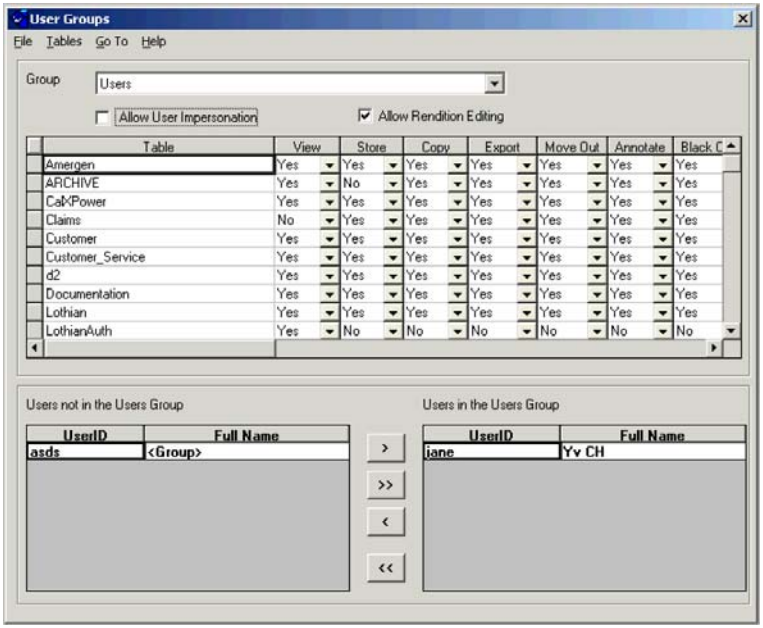
2 **Select how users and groups are defined (*Outside of Documanage* or *Within the Documanage Database*) for your Documanage System.**

If you select *Outside of Documanage*, the ***User Groups*** dialog for the NT LAN Manager appears.

.



If you select *Within the Documanage Database* the *User Groups* dialog for the Database appears.



3 **Select the user group that you want to assign security privileges to from the *Group* list.**

The users that are in the group appear on the *Users in Group* list, User group information displays, including tables and their annotations and other document-management rights.

The *Allow User Impersonation* check box becomes available, and the Documanage User IDs and the Full Names of the members of the selected group are listed in the *Users in Group* list at the bottom of the dialog.

NOTE: Group Names can contain up to 64 characters. Restricted characters for User and Group names within the Documanage Database are / \ | , ? * “.

- To have sessions based on members of the selected group impersonate other users, select *Allow User Impersonation*.

NOTE: User Impersonation allows you to impersonate another user during a transaction without storing the user’s password. Select the Allow User Impersonation checkbox to control which groups can impersonate other users. This keeps a user without a specific name and password from writing an application that can impersonate another user. This would allow such a user to do things like viewing another user's diaries.

- To allow users in the group to create and manipulate renditions, select *Allow Rendition Editing*. You must also set the *See History* authority for the appropriate tables in the dialog to *Yes*.

NOTE: Using renditions, files containing the same content but with different formats can be automatically interchanged to meet the requirements of different applications. For instance, with renditions it is possible to: display an image on a monitor using a TrueColor file; print the image on a black and white printer from a monochrome (or grayscale) file; and incorporate the image into a pre-print color-separation from a four-color file.

- If you are using the Documanage database access model and want to add members to a group, select the users and/or groups to be added from the *All Users Not In Group* list. Click the right arrow button to move them

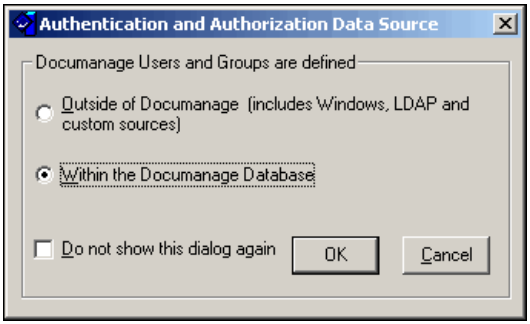
into the *Users in Group* list. Click the double arrow buttons to move all the users and groups.

- If you want to remove members from a group, select the users and/or groups to be deleted in the Users In Group list, and click the left arrow button to move them out of the group. Click a double arrow button to move all the users and groups between lists.

- If you are using the Documanage database access model and want to add a new group:
  - Select *NewGroup* from the *File* menu.
  - Enter the group name into the dialog and click *OK*.

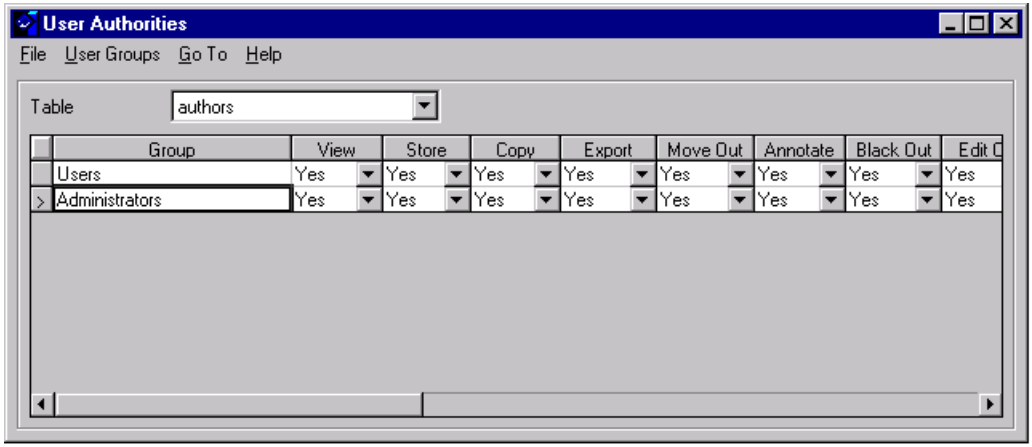
To assign security via the User Authorities dialog:

1. **Click the *Authorities* button on the main Documanage Administrator dialog.**
   The *Authentication and Authorization Data Source* dialog appears.



2. **Select how users and groups are defined (*Outside of Documanage* or *Within the Documanage Database*) for your Documanage System.**

The *User Authorities* dialog appears.



3 **Toggle the appropriate security attributes either to *Yes* or *No*.**

4 **If you want to add a group to the Group column:**

- Select *Add Group* from the User Groups menu.
- Select the group you want to add from the *Group* drop-down list.

5 **If you want to delete a group from the Group column for the selected table:**

- Highlight the group to be deleted.
- Select *Delete Group* from the User Groups menu. A confirmation dialog displays.
- Click *Yes*. The group is deleted.

6 ***Save* and *Exit*.**

Security attributes set and saved in the *User Authorities* dialog will also appear in the *User Groups* dialog.

To give users the authority to edit only certain documents in a folder

1. **Click *Start|All Programs|Authorities***

   The *Authentication and Authorization Data Source* dialog appears.

2. **Select how users and groups are defined (*Outside of Documanage* or *Within the Documanage Database*) for your Documanage System. Click *OK*.**

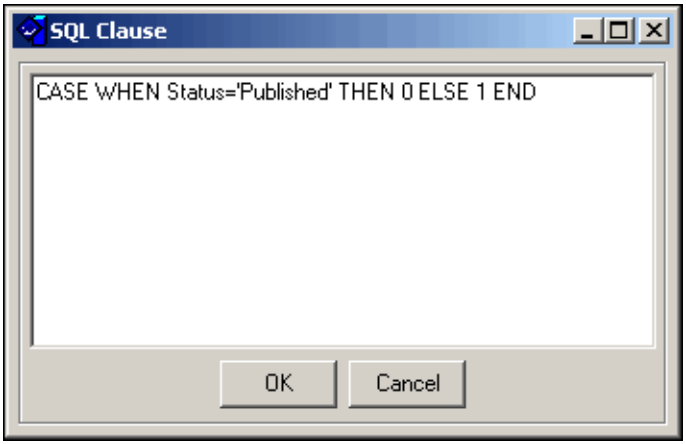
   The *User Authorities* dialog appears.

3. **From the *Table* list, select the database table on which your cabinet is based.**

4. **Select the *Group* that to you wish to award the conditional edit authority.**

5. **Select *SQL* from the *Edit Contents* list.**

   The *SQL Clause* dialog appears.



6 **Type an SQL SELECT expression that returns a 1 (TRUE) if the document can be edited by the group, and a 0 (FALSE) otherwise. Click *OK*.**

The *SQL Clause* dialog closes.

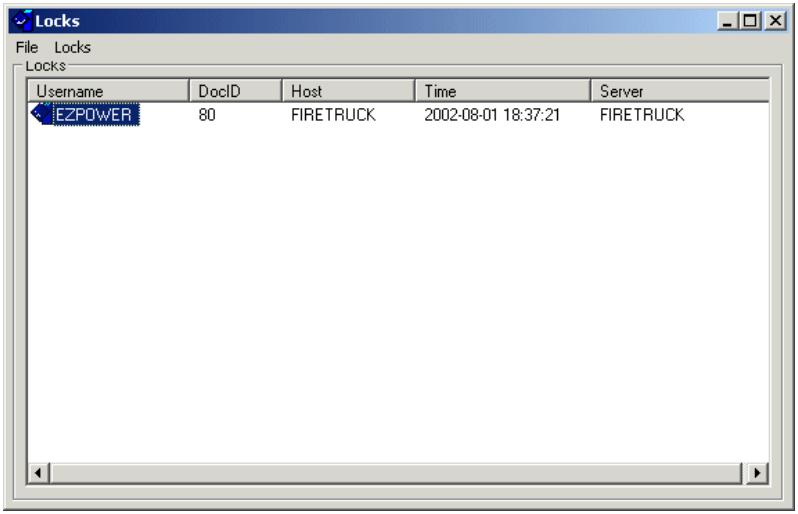
For example, the following SELECT expression gives members of a group Edit Contents authority for unpublished documents, but not published ones:

CASE WHEN Status='Published' THEN 0 ELSE 1 END

7 **Close the *User Authorities* dialog.**

Locks

When a user is annotating a document in Documanage Workstation, that document is placed in annotation lock mode. The locked document's properties display on the *Locks* dialog in Documanage Administrator so that an administrator can view and remove locks if necessary.

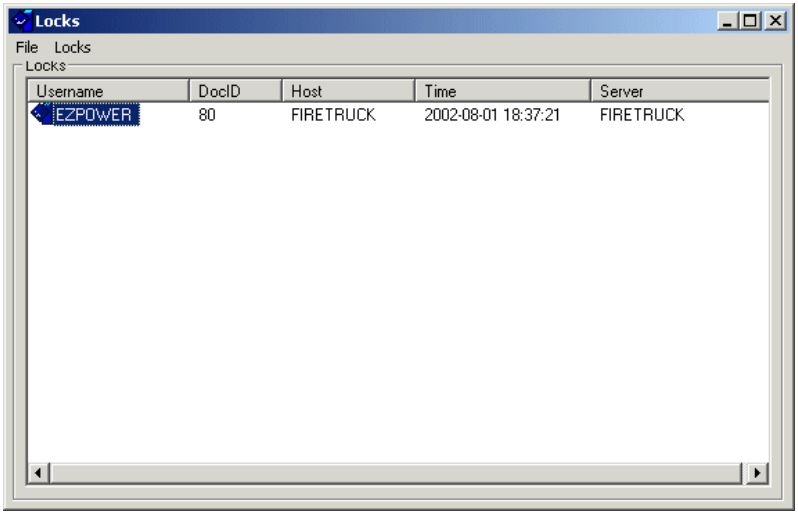


An administrator might delete locks if the system crashes and certain files are not released by the server. The administrator can then go in and delete the locks so the file is available for use by everyone with access.

To view and delete locks

1 **Click the *Locks* button from the main Administrator dialog.**

The *Locks* dialog opens and displays the *Username*, *DocID*, *Host*, *Time* and *Server* of the files that are locked.



2 **Select the lock you wish to delete (if you wish to delete a lock) and go to *Locks|Delete Lock.***

The selected lock is deleted.

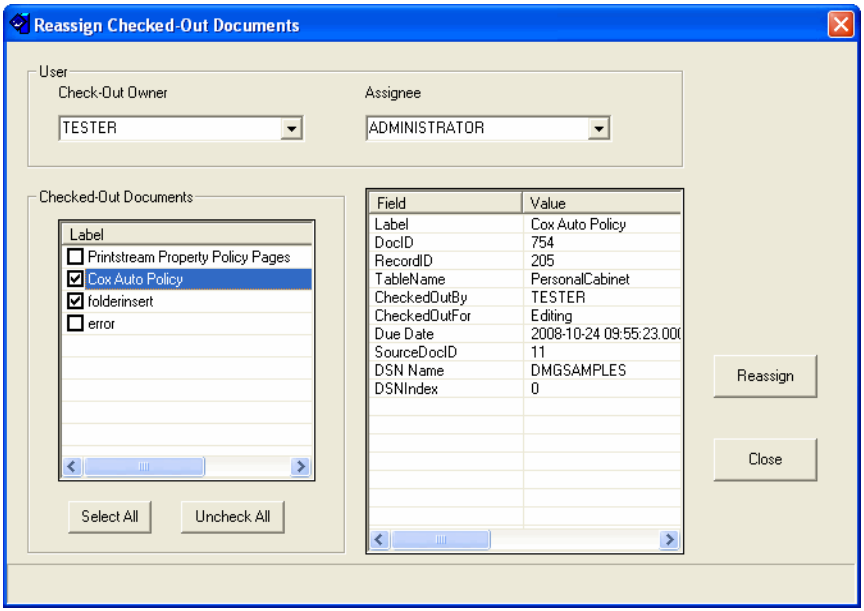
3 **Exit the dialog when finished.**

Reassign Checked-Out Documents

A document checked-out to a specific user can only be checked-in by that same user (referred to her as the “owner”). If the check-out owner is no longer available, an option has been provided to reassign the checked-out documents to another user (referred to as the “assignee”).

To reassign checked-out documents

1 **Click the *Reassign Check-Outs* button from the main Administrator dialog.** The Reassign Checked-Out Documents dialog is presented.



2 **Select a check-out owner from the picklist.** The checked-out documents owned by that user will be presented in the list**.**

3. **Select an assignee from the picklist. The checked-out documents will be reassigned to this user.** Only users of Documanage that have a Personal Cabinet defined will be presented in this list. (A user will have a configured Personal Cabinet if they run Documanage Workstation)**.**

4. **Check the documents to be reassigned.** Use the “Select All” and “Uncheck All” buttons to toggle the selection of all documents in the list. Information about the current document is presented in the right-hand list**.**

5. **Click the “Reassign” button to reassign the selected documents.** An informational message will display after the documents have been reassigned.

6. **Select a new owner to reassign or click “Close” to close the dialog.**

Diagnostic Tests

Documanage Diagnostic Tests allow other Oracle products to have a Documanage server run tests that assess the accessibility of the storage volumes and database used by that server.

Diagnostic tests are defined in the poffice.ini file, and each one must be given a unique name of up to 255 characters. Each diagnostic test is a section of the INI file and has one or more runtest statements that test the storage volumes and database. Each runtest statement has a type (listed in the table below) that determines the specific nature of the test to be run. There can be no more than one instance of any type of runtest statement in each diagnostic test.

The syntax for the section heading of a diagnostic test is:

```
[DIAGNOSTICTEST:<name>]
```

where <name> is the name of the diagnostic test. This section heading is followed by a list of runtest statements. The runtest statements are executed in the order in which they are listed. Each runtest statement has the following syntax:

```
RUNTEST(<type>)=<parameter string>
```

where `<type>` is one of the types listed below and <parameter string> is appropriate to the particular type.

<type>	<parameter string>	Description
FSMIN	Comma separated VOLUME list or the keyword _ALL_ for all VOLUMES	Verifies access to specified VOLUMES
FSREAD	Comma separated VOLUME:<filename> list	Opens the specified file in the volume and reads from it.
FSREADWRITE	Comma separated VOLUME list or the keyword _ALL_ for all VOLUMES	Creates a temp file in the specified VOLUME, writes to and reads from the file and deletes the file
DBSQL	Comma separated DSN:<SQL> list and use the keyword _ALL_ for all DSNs	Executes the specified SQL against the specified DSN. *Only select statements are executed, others are ignored*
DBCON	Comma separated Data Source names or the keyword _ALL_ for all DSNs	Verifies connectivity to specified database(s)

Remember: Each type of runtest statement may be used at most once in a diagnostic test. For example, if access to multiple volumes is to be tested, all volume names should be listed in a parameter string to a single FSMIN runtest statement, and not as multiple FSMIN runtest statements.

In the following examples, notice that the keywords are not case sensitive.

Example 1:

```
[DiagnosticTest:CheckFSandDB]
RUNTEST(fsmin)=VOL1, VOL2
RUNTEST(dbcon)=DMANAGE, pubs
```

This diagnostic test is named "CheckFSandDB". It has two runtest statements. The first verifies the connection to the volumes named "VOL1" and "VOL2". The second verifies the connection to the databases named "DMANAGE" and "pubs".

Example 2:

```
[diagnostictest:CheckFSandDBall]
RunTest(FSMin)=_ALL_
RunTest(DBCon)=_ALL_
```

NOTE: When using the parameter for ALL, the underscore characters are part of the required parameter.

This diagnostic test is named "CheckFSandDBall". It has two runtest statements. The first verifies the connection to all volumes. The second verifies the connection to all databases.

Example 3:

```
[DIAGNOSTICTEST:CheckFSandDBadv]
runtest(FSREADWRITE)=VOL1, VOL2
runtest(FSREAD)=VOL3:myfile1.jpg,VOL4:myfile2.met
runtest(DBSQL)=pubs:<select * from
   employee>,authors:<select * from address>
```

This diagnostic test is named "CheckFSandDBadv". It has three runtest statements. The first verifies read/write access to the volumes named "VOL1" and "VOL2". The second verifies that the file myfile1.jpg on VOL3 and the file myfile2.met on VOL4 can be read. The third verifies that a SQL statement can be run against the employee table of the pubs database and the address table of the authors database.

Web Workflow Designer

Introduction to the Web Workflow Designer

NOTE: This feature is not available with the Documanage Server Lite product.

Documanage Workstation provides routing of documents along a pre-defined series of destinations (i.e., people or departments) in an organization. *Documanage's Web Workflow Designer* module enables you to design a workflow map which contains the steps needed to route documents for a particular project. The new thin-client Workflow Designer tool provides complete flexibility in handling workflows from the very basic to the very complex, depending upon your organization's needs.

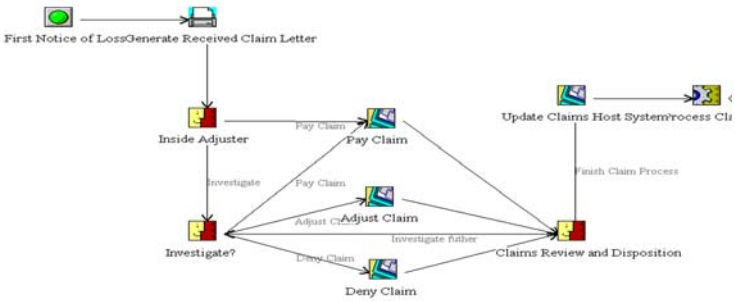
In this chapter

The information in this chapter will help you answer the following questions:

- What are workflow maps?
- How do I use the Web Workflow Designer to create workflow maps?
- How do I use workflow maps in projects?

Understanding workflow maps

A *workflow map* is a set of *tasks* and *links* which form the complete steps needed to route documents. A task is a destination where the documents are routed to or from. A link is the connection from one task to another. A map is processed by advancing from task to task along each link connecting them together. A workflow design may originate as a diagram detailing all the steps a workflow must complete to process. An example of such a diagram is below:



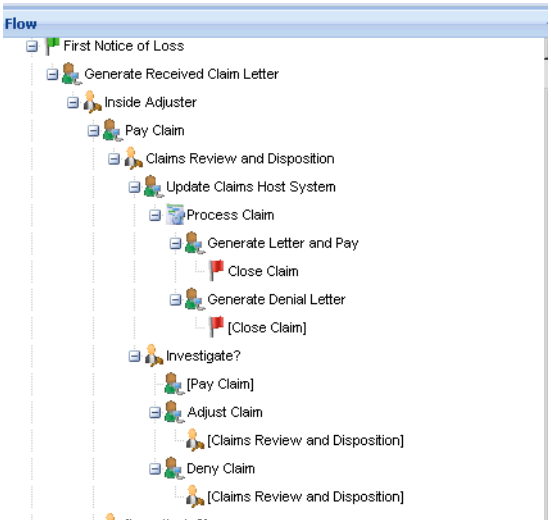
Every workflow map is based upon data from a table in a database. The table that is used is called the ***workflow table.*** This table can be chosen from among all the tables that are mapped to Documanage. The documents in a workflow are associated with data in the workflow table (so, in a claims processing workflow the documents in the workflow might be associated with the data from the customer ‘Claims’ table), to create workflow folders called *projects*. Documanage implements data driven workflows with the association between the data in the table and the documents.

For example, a workflow designer wants to create a way to route claims for processing through all staff that must approve the claim. The designer, or *administrator*, of the workflow must create a map linked to the customer

database table that has data relevant to claims processing: this could be a table called ‘Claims’ in the Customer database.

Claim_Number	Vehicle_ID	Claim_Remarks	POWER_Tag	Policy_Number	Claim_Approved
123534864898	7	Rear Bumper	49	200002587785	YES
123589887546	10	Front Bumper & Cor	50	200110225874	NO
125256488481	12	Rear End	51	199906105486	YES
132554988868	1	Rear End	53	199905143580	YES
153548498735	14	Rear Right Bumper &	54	199805152548	YES
154543543548	24	Driver's Side Front a	55	200203101576	YES

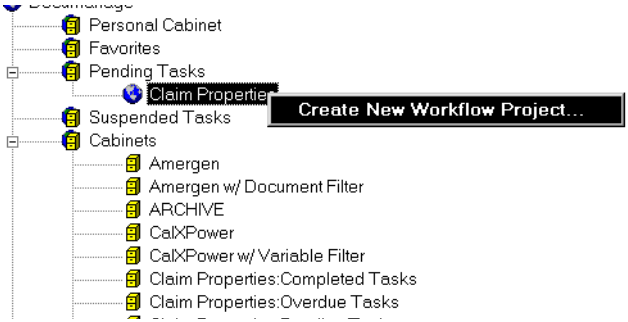
The table is mapped to Documanage and then used to create a new workflow map based upon the original analysis represented in the diagram. The workflow administrator creates tasks that must be completed to route a claim. The administrator assigns a manager and a team of users to each task. The tasks are linked together to complete the map. In our example, the completed Claims map could appear as this one below:



Once the workflow map is ready for use, the administrator must release it. A map is actually made available for processing by ***activating it.*** When a workflow map is ready for use, an administrator ***releases it***, which automatically activates it. If there are no users working in the workflow, it can be ***unreleased*** to make changes, but if any work is being done in the workflow this will result in possible data loss. If editing is required on an active workflow map, a safer option is for a manager to use the ***checkout*** feature, making a locked copy of the workflow map that safely allows work to continue on the original workflow.

Once changes are complete, an administrator then uses the ***checkin*** feature on the altered version of the workflow. This checkin process will overwrite the original workflow without loss of any data that is currently being processed. A released map can also be ***deactivated,*** or set so that no further processing can be done on it. A deactivated map can be reset for processing by ***activating*** it.

In our Claims Processing workflow example, the users assigned tasks in the workflow would then be able to start projects, and create new folders, within the tasks assigned to them. Below is a view of how this might appear to users in Documanage Workstation:



The users could now import documents into these folders, based on the structure of the ‘Claims’ table. Once a user has accepted, or checked out, a

task and worked on the document in that folder, they can advance the folder to the next task within the workflow, or choose to check the project back in for use by another member of their user group.

Workflow roles

There are several roles in the workflow model:

- **Administrators**. Administrators are responsible for designing and maintaining a workflow map. An additional duty is monitoring a workflow for adequate throughput and lack of bottlenecks. All Administrators are members of the *Documanage_Managers* group.

- **Managers**. Managers are responsible for assigning teams of people to various tasks. All managers are users whose names are taken from NTLM, LDAP, or database security. Only one manager is assigned to each task.

- **Team Members**. Team members are responsible for performing the various tasks in a workflow. Each team consists of one or more groups taken from NTLM, LDAP, or database security.

Getting started

The Web Workflow Designer is used to create the workflow map containing the steps needed to route documents for a particular project. The Web Workflow Designer allows you to:

- Create maps
- Create tasks and link them together
- Assign managers and team members to each task
- Create Workflow Task Containers
- Release maps
- Activate and deactivate maps
- Checkout and Checkin workflow maps
- Delete maps.

Before you can use the Web Workflow Designer to create a workflow map, you must start and log in. Note that your start-up procedure may differ slightly from the one below.

To start the Web Workflow Designer

1. **Launch the Web Workflow Designer. (Usually a url set up in installation, such as *http://your computer name:8080*).**

   The *Authenticate User* login dialog displays (allow pop-ups in your web browser).

2 **Type your user name in the *Name* text box, then fill in the *Password* and *Domain*.**

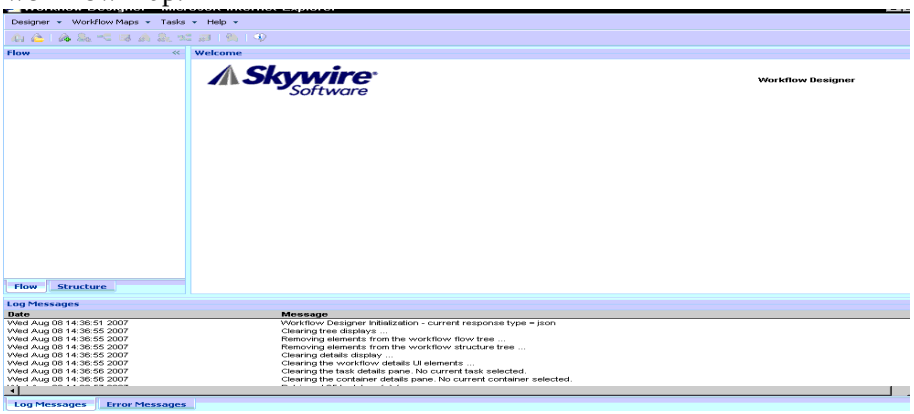
NOTE: Web Workflow Designer validates the user name and password from the Documanage Router, utilizing NTLM, LDAP, or database security.

To exit Web Workflow Designer

1 **Choose *Designer|Exit* from the File menu.**
Web Workflow Designer quits the application and returns you to Windows with a confirmation message.

Using the main window

Web Workflow Designer has several features that help you to create a workflow map.



- **Menu bar.** Menus provide access to commands that you use to create a workflow map. Some commands allow you to initiate actions immediately. Others display a dialog, which lets you specify file locations and settings and then initiate actions. Commands that are not available are dimmed.

- **Tool bar.** The tool bar offers an alternative way to initiate standard commands. If a function is not available, the corresponding button on the tool bar is dimmed. Placing your cursor over an item in the tool bar will show a tool tip for that icon.

- **Details/Properties area.** The right section of the Web Workflow Designer is the details or properties window. The details on the workflow maps, and the properties for links and tasks display here.

- **Workflow Map views: Flow vs. Structure**. The workflow map area, on the left side of the screen, has two view options, selected by tabs at the bottom of the pane. The *Flow* tab will show a picture of the flow of workflow tasks within a map; it shows a visual display of how tasks are linked to one another. The *Structure* tree shows tasks grouped according to *Task Containers*. (For very complex workflows, it is sometimes easier to view tasks, and their links, in the Structure tree, instead of the drill down view in the Flow pane.)

- **Log messages/Error messages**. These two tabs appear at the bottom of the Designer, under the Status bar, and display general log messages by default. If you encounter errors, such as an error during validation, the Designer will warn you of an error, then automatically toggle to the Error message tab, where the error will display.

These messages are different from the web server's log and are not saved in any file. Messages are lost when the user session in the Web Workflow Designer ends.

Date	Message
Fri Aug 03 12:45:08 2007	The Task - trsdt failed validation:
Fri Aug 03 12:45:09 2007	Task is invalid: [The task has no route to the 'END' task]

Log Messages | Error Messages

Creating a map

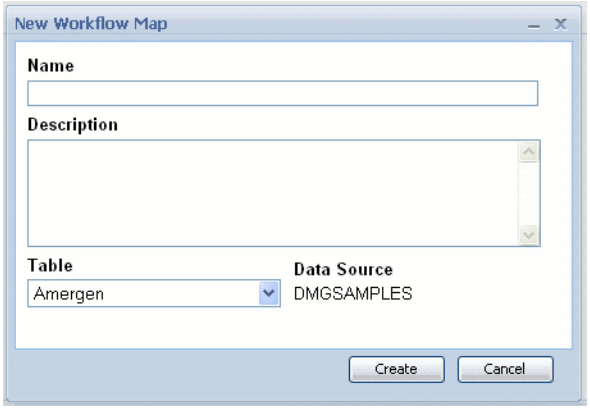
After you've started the Web Workflow Designer and logged on, you are ready to create a workflow map. You can create a new map from scratch or by importing the properties of an existing map, using the *Import* menu item. When importing an existing map, all the links and tasks are kept intact, requiring minimal effort from a system administrator. A map may be *exported* for use later if you wish to build another, similar map at a later time.

NOTE: It is recommended that you perform analysis on a workflow design prior to beginning map creation in the Web Workflow Designer, especially for very large or complex workflows. This way, you can utilize features, such as *Task Containers* and the *Structure* tree grouping of tasks, to your advantage.

To create a new map

1 **Select *Designer|New|Workflow Map*, or the *New map* tool bar button.**

The *New Workflow Map* dialog displays.



2 **Type a map name in the *Name* text box, then press <Tab> and add a Description.**

Enter a unique name up to 32 characters long. The workflow name should be descriptive of the process it facilitates. You can use spaces to improve readability. Duplicate map names and single quotes are not permitted.

Enter a description up to 100 characters long that tells users who are unfamiliar with the workflow what its purpose is

3 **Choose a table to base the map on from the *Table* drop-down box.**

This box includes all tables or views which are currently mapped to Documanage. When a table is chosen, the database in which it exists is automatically displayed in the *Database* text box. The table you choose must contain the data that you want to associate with the documents.

4 **Click *Create*.**

The *New Workflow Map* dialog closes, the Begin and End tasks are available, and the workflow details window displays properties of the new workflow map. You are able to select completion options for the workflow projects on this screen. These include the following:

- *Delete the documents in the project*. When checked, indicates that documents in the current workflow project will be deleted upon completion of the END task.
- *Delete the folder*. When checked, indicates that the database table folder corresponding to the current workflow project will be deleted upon completion of the END task.
- *Delete the Project*. When checked, indicates that the entire workflow project, including folders and documents, will be deleted upon completion of the END task.

To Delete a Workflow Map

1 **Select *Designer|Delete|Workflow* Map, or use the *Delete workflow map* button on the tool bar.**

The confirmation message appears.

2 **Click 'Yes' to confirm. The Web Workflow Designer window will refresh and the map will be deleted.**

To open an existing map

1 **Select *Designer|Open*, or click the *Open workflow map* button on the toolbar.**

Opening a workflow will automatically close any currently open workflow.

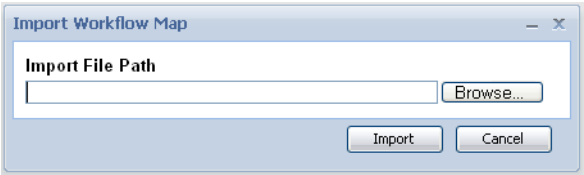
2 **If you wish to save the existing map as a new map, select *Workflow Maps|Export*.**

The *Export* dialog window appears, and you will prompted to save the map as an XML file. Once this is saved, you can then use the Designer, Import menu item to import the map and build another workflow.

3 **To Close a Workflow Map, select *Designer|Close* from the menu bar.**

4 **To Import an existing Map, select *Designer|Import* from the Menu bar.**

The *Import* dialog will appear. Browse for the xml file to import:



Now that you have created or opened a map, you are ready to add tasks to your newly workflow.

Working with tasks

The first step to perform after creating a workflow map is to add tasks. In order to add tasks properly, it is important to know what task types are available.

There are two system tasks required in a workflow map: “BEGIN” and “END.” There can only be one of each in a map. All the tasks of a map must be linked from the BEGIN task to the END task. The system automatically creates these tasks. They cannot be added to or deleted from a map.

- **BEGIN:** This task is the starting point of all maps. The team members assigned to this task are able to create projects.

- **END:** This task is the final task of a map. No team members can be assigned to this task.

In many instances, the route that a map takes is dependent upon the action taken in a previous task. These tasks are referred to as decision branches. There are two types of decision tasks from which you can choose: automatic branch tasks and human decision tasks.

- **Automatic branch task:** This task automatically routes a project to one of the tasks linked to it. It is routed to another task based upon the data in the workflow table. For example, in the Claim Processing workflow, if the adjuster approves the claim, a letter is generated along with a payment flag. Therefore, *Claim Approved = Yes* routes all documents in folders marked with a value of "YES" in the *Claim_Approved* data field to the payment department, while all other values route the workflow to a denial letter task type. This step is automatically accomplished, precluding the need for human intervention.

  To use an automatic branch task, you must create a "rule" for every link from the automatic branch. These rules are based on the columns in the database table upon which the workflow is based. The links leaving an automatic branch task can have only a single DEFAULT rule. A workflow cannot be released if the Task Link properties of one or more of the links leaving its Automatic Branch tasks are set up incorrectly. A map advances to the link with the DEFAULT rule if the conditions set in any other rule are not met.

  An example of the automatic branch rules, using the Claims Processing example above, might be *Claim_Approved='Yes'*, and *DEFAULT (*for any other claim status.) See "Working with links" on page 167 for details on adding a link rule.

NOTE: Creating proper rules for an automatic branch is important in order to ensure that the map is routed correctly. The rules in the links from an automatic branch do not have to be fully qualified SQL statements, but use a single quote (‘) around the data value to be evaluated. If you are unsure of how to create a rule, contact your Database Administrator for more details.

- **Human decision task:** This task gives the team member processing it the ability to route a project to one of the tasks linked to the human decision. When the task is processed, the team member chooses the appropriate task to send the project to. For example, a claim documents would be sent to an adjuster for review. That user could then deny or approve the claim. The team member can not choose more than one task to route a project along.

- **Consolidation:** A map can be set up so that a task will not advance to the next task until all the tasks which are linked into it are processed. This type of task is a consolidation task. A map will not be advanced to a consolidation task until every parallel task is processed. For example, a manager may want to review all documents together for a claim before giving final approval to the project. Until all documents are approved, the project will not be advanced.

- **Auto Mail**: An Auto Mail task is a special task which tells the Documanage Server to automatically send mail to a user while this task is processed. Unlike other tasks, this task will not show up in the user’s Inbox.

Any tasks, which are not part of a category listed above, are considered **Work**, or generic, tasks. Whenever an operation, such as adding a document to a file, needs to be performed a generic task can be used.

To add a task

1. **To add the first task, you must first click on the existing BEGIN task, and then edit the properties of this task. You must assign one Manager, and then select a user group to work on the task. This group will be able to start a workflow project. Add a description to the task as well (***see step 4***).**

2. **Now you may add another task type. First, click or highlight the task you wish to link the NEW task to.**

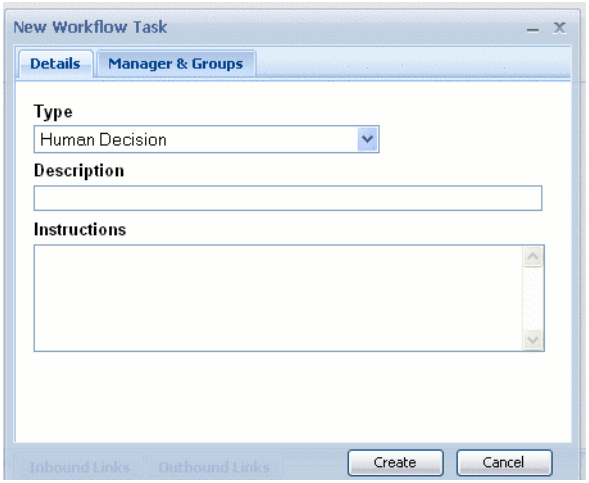
   Then either select *Designer*|*New*|*Workflow Task* from the menu, or use the *New Workflow Task* button on the tool bar.

3. **The *New Workflow Task* window appears. Select the type of task from the drop down list.**

4. **Then fill in the description.** Every task is required to have a description, which is what you see for the task when looking at the Web Workflow Designer window. It is also what a team member sees when they are processing the task. Each description should be unique, since this helps teams to process tasks.

5. **What is written in the *Instructions* is what assigned users will be able to read when they accept the task in the workflow project.**

   This is an optional property. This property is used to help a team member when they are processing a task. If they need instructions on a task, they can click a *Help* button available in the Workstation and display the instructions entered here.

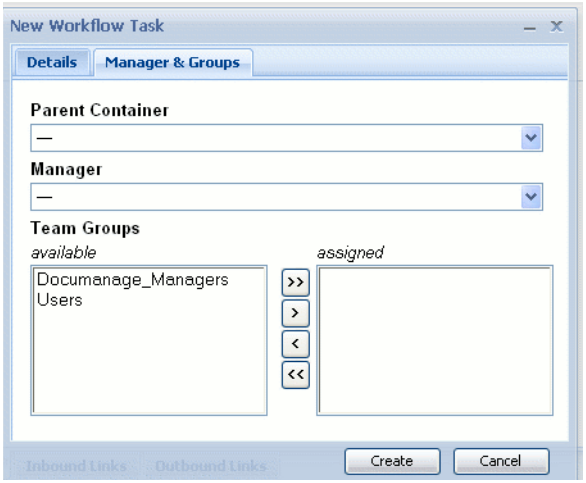


6 **If you click 'Create' now, you will be prompted to assign a Manager, located on the '*Managers and Groups*' tab.**

Every task must have a manager. The manager of the task is the person who is responsible for assembling each team to fill each role. A manager can manage more than one task. The team assigned to a role can be changed at any time.

7 **Add at least one Team Group (users group) to the task.**

Every task must have a team. The members of the groups on the team given to a task can process the task.



NOTE: Missing required items (Manager, Team Groups, and Description) are flagged in red.

8 **Now you will be prompted to save changes to the workflow map.**

9 **If you click on the Inbound Links tab in the workflow task details window, you will see that the task you clicked on originally (in our example, the BEGIN task) shows up as the task that links inbound to the new task. We will look at adding Inbound and Outbound links in the "Working with links" on page 167 section of this chapter.**

To delete a task

1 **Highlight or click on the task you wish to delete.**

2 **Then either select *Designer|Delete|Workflow Task* from the menu bar, or click on the *Delete workflow task* button on the toolbar.**

This deletes the task and all of its links.

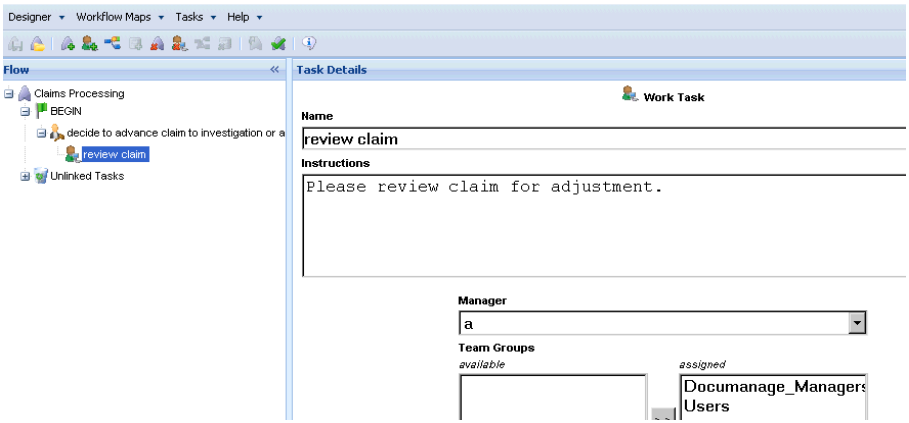
To change a task's details

It is necessary to specify task properties in order for a task to work properly within a workflow.

1 **If a task has already been added to the workflow map, click on the task in the map.**

2 **The Properties or details of the task will appear on the right hand pane of the Web Workflow Designer.**

The *Task Details* dialog corresponding to that task displays.

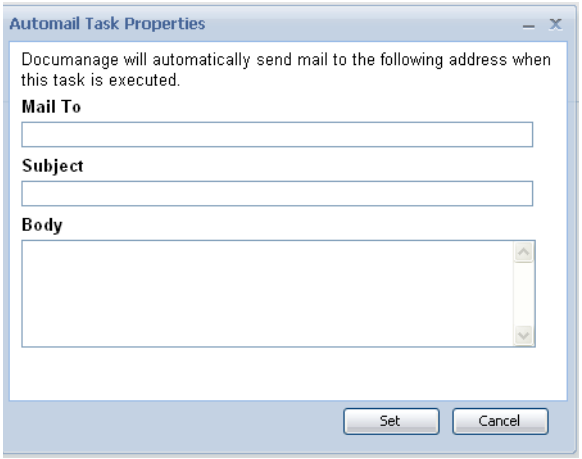


3 **To change the task properties, including name, instructions, manager or team, type in the new information or select new options from the drop down lists.**

Every task must have a name, which is the type of task being used. This property can be changed, except in the case of BEGIN and END tasks.

4 **If you wish to change the email properties on an Automatic Email task type, first highlight or click on the task in the workflow map. Then, select *Tasks|Automail Details* from the menu bar.**

The *Automail Task Properties* dialog displays.



An Auto Mail task is a special task that tells the Documanage Server to automatically send mail to a user when this task is processed. Unlike other tasks, this task will not show up in the user's Inbox.

NOTE: To use the Auto Mail task the Documanage server must be set up to mail.

To Validate a Task

All tasks in a workflow map must be valid for the workflow to process successfully. If a task is not valid, the workflow map will not release. A task is valid if it meets the following criteria:

- task has a manager
- task has one or more team groups (if it is not an END task)
- task has at least one route from BEGIN type task to END type task
- all Automatic Branch outbound links have a non-empty and valid RuleSQL
- all Automatic Branch outbound tasks have one DEFAULT RuleSQL outbound link
- all Human Decision tasks links have a non-empty ‘Decision’ caption
- All Automail task types have a non-empty ‘Recipient’

1. **The workflow must either be unreleased or be a checked-out copy to validate a task. Highlight or click on the task you wish you validate from the Flow or Structure pane, then select *Tasks|Validate* from the menu.**
2. **The Task Validation Complete dialog appears. If the task fails validation, the reason will post in the Error Message pane.**

Working with links

The next step to perform after creating a workflow map and adding tasks is to add “links.” Links connect one task within a map to another. Every task must be part of a link from the BEGIN task to the END task. When a task is processed, the project is routed to the tasks to which it is linked.

In order to add links properly, it is important to know what link types are available. There are two types of links used in Documanage: “*Inbound*” and “*Outbound*” links. The functionality of these links is the same. Inbound links display on the Inbound Links tab in the details of a task; all tasks linked to the task you are viewing show up in the Inbound links table. If the task you are currently viewing is linked to other tasks, these links display on the Outbound Links tab.

Links may have certain properties depending upon the task types they are connecting. For example, Automatic Branch task types have Outbound links that reflect SQL statements for routing a folder. Human Decision tasks have a ‘Decision’ parameter for Outbound links.

To add a link

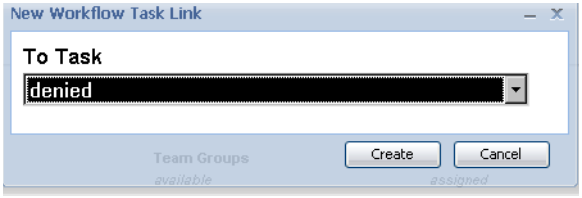
There are several ways to add a link.

1. **Inbound Links are created when a new task is added to the Workflow Map. When adding a new task, you must highlight an existing task (such as the Begin task) and then add a new task. This automatically creates an Inbound link from the original task to the new one.**

2. **To create an Outbound link, you must select a task and then click on the Outbound Link tab in the Task Details window. Then select either *Designer|New|Workflow Task Link* from the menu, or use the**

    ***Add Link* button on the tool bar to create a new link.**

3 **The *New Workflow Task Link* dialog appears.**



4 **Select the task to link to, then click Create.**

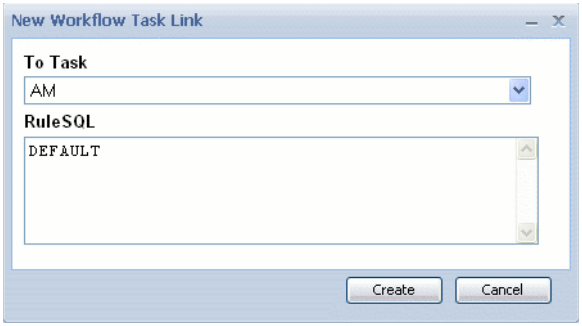
The Task Details window now displays the Outbound link task that was just added.

Outbound links for Automatic Branch or Human Decision tasks

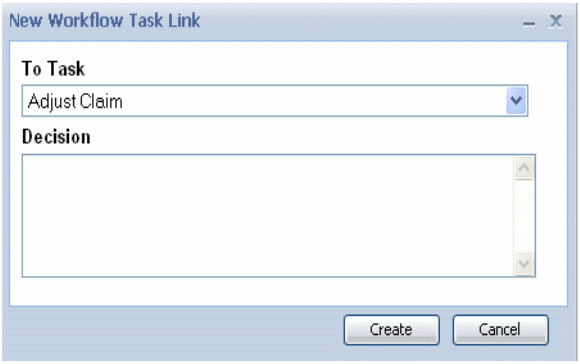
When adding an Outbound link to an Automatic Branch or Human Decision task type, you can specify properties of the link. Links can be created to any tasks that are *unlinked* to the Workflow Map, listed under “Unlinked Tasks”, or to any existing task.

1 **An Outbound task from an Automatic Branch will need a ‘Rule SQL’, which, if TRUE, is the SQL statement that the branch must follow to route a folder in the Workflow Map.**

An Automatic Branch should have at least two Outbound links: one for the specific SQL rule that must be met, and one for all other values (‘DEFAULT’).



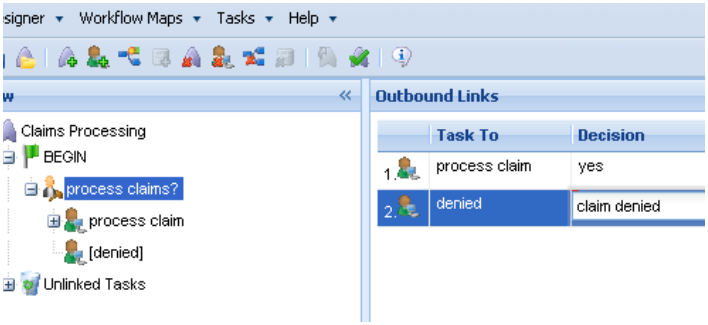
2 **A Human Decision task should have at least two Outbound links. User Teams working on this task will see 'questions' when advancing these task types. The questions they see are what the Workflow administrator types in the Outbound link 'Decision' window.** *(The 'Decision' pane allows up to 20 characters.)*



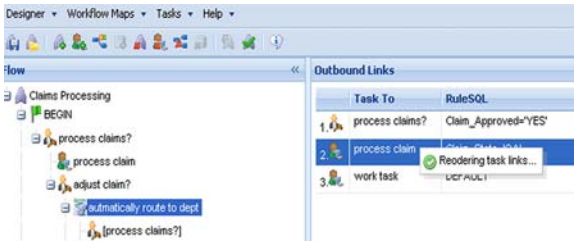
3 **The Outbound Links tab for both Automatic Branch and Human Decision task types have an extra column in the link table that displays the properties of the links.**

If a Workflow Map is not released, or is checked out, these columns are editable. A Human Decision task link will allow manual editing, and an Automatic Branch link will launch the SQL Builder.

Please see “Automatic Branch links using the SQL Builder” on page 171 for details.



Outbound links of Automatic Branch tasks are processed in the order listed. If needed, these can be reordered once they are added to the Outbound Links tab of an Automatic Branch task type. The links can be highlighted, then dragged and dropped into a new order from this window. *Note that the ‘DEFAULT’ links is always last, and cannot be reordered*.



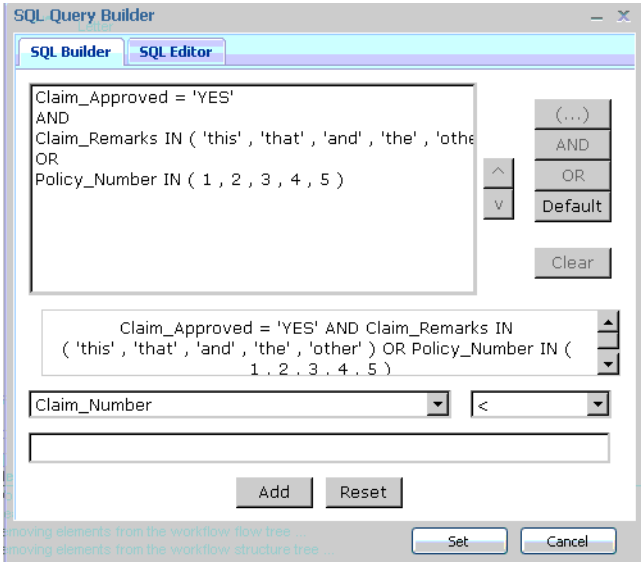
Automatic Branch links using the SQL Builder

To reduce the possibility of errors when creating an Automatic Branch Rule SQL, you can use the SQL Builder. This is an interface for adding, editing, deleting, and grouping clauses. It also provides a selection of operators based upon the data type of the column used in the rule. This interface also allows manual editing of the SQL Rule.

The following steps describe how to access the SQL Builder:

1. **Add an Outbound link to an Automatic Branch rule, following the directions above in “Outbound links for Automatic Branch or Human Decision tasks” on page 168.**

2 **Double-click on the RuleSQL column on the Outbound links tab of the new task. A drop-down arrow will appear at the end of the column. To access the SQL Builder, click on this arrow, and the SQL Query Builder dialog opens:**



The SQL Builder tab is the interface for adding fields and operators from drop-down tables to build a SQL Rule. The Rule is displayed in the box above the *column*, or field, table as you work. The *components list* window at the top of the Builder pane displays all distinct clauses utilized in the current Rule, and allows you to re-order, group or delete clauses from the Rule.

The SQL Editor tab allows manual editing of the SQL Rule; for example, if you may use the SQL Builder pane to construct a Rule using column elements, and then add a custom script to this Rule that is not based on database fields. Changes in the Builder pane will reflect in the Editing pane and vice versa.

3 **To add a field to the SQL Rule, click on the drop-down arrow in the first box and select the field.**

4 **Operators will vary based upon the data type of the selected column (ex., a numeric field will provide different operators than a date field). To add an operator, click on the *operator* list next to the column selection window.**

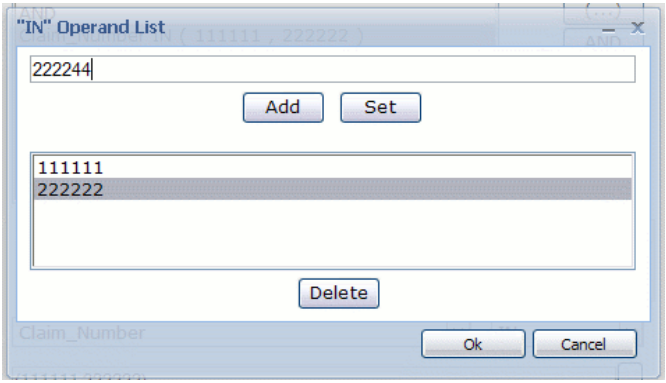
Operators include:

operator	description
>	greater than
>=	greater than or equal
=	equal
<>	does not equal
<=	less than or equal
<	less than
IS NULL	the column is null
IS NOT NULL	the column is not null
IN	one of a list,
LIKE	the column's data contains, begins with, or ends with specified values
BETWEEN	the column's date value is between the provided start and end dates

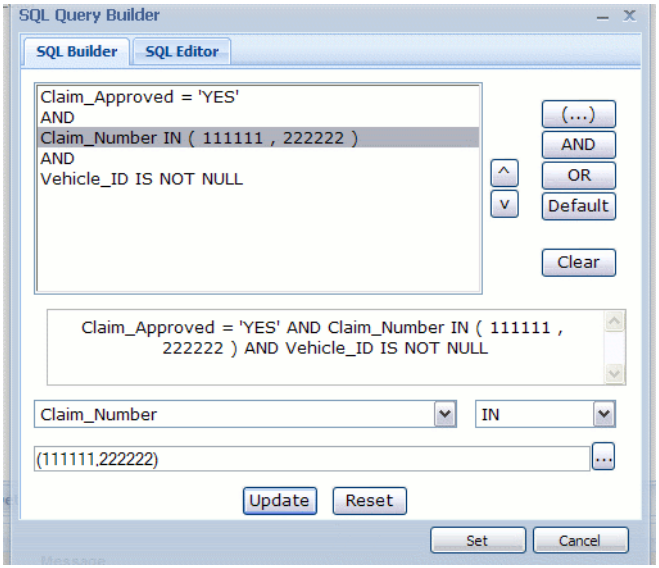
5 **Once an operator has been selected from the list, the user will supply the data to be processed by the rule in the *value fields* box at the bottom of the Builder pane.**

If the operator is ***IN***, the user can click on the edit button at the end of the value fields box to open the *Edit field* window. This window will let the user Add values for a field, and then the SQL Builder will correctly enter the values to match their date type in the database, such as ODBC, date or numeric vs. string syntax.

The *Set* button will edit and change selected values in the list. The *Delete* button deletes a highlighted value. Clicking *OK* saves the entries, while clicking on the *Cancel* button takes you back to the Builder pane without saving any values.



6 **Once you ar back in the Builder or Editor pane of the SQL Builder, to add a new clause created from the field and operator drop-down tables, click on the '*Add*' button.To re-order or group individual clauses built using the steps above, click on the line, or clause, in the list in the uppermost box of the Builder pane that you wish to work with, then use the *arrow, (...), AND,* or *OR* buttons to edit the statement.**



7 **If you have edited an existing clause, an '*Update*' button will display instead; click on this to update the existing clause.**

8 **To remove a line from an existing clause without deleting the entire Rule, select the clause in the list, and then click on the '*Clear*' button.**

9 **To complete the sql statement and exit the SQL Buider, click on the '*Set*' button on either the Builder pane or the Editor pane.**

10 **To exit the SQL builder without setting any changes, click on the '*Cancel*' button. You will be asked if you wish to discard any changes you made to the SQL Rule: click 'Yes' to clear changes. Clicking 'No' will put you back in the SQL Builder where you can Update and Set any changes you made to the Rule.**

11 **The RuleSQL column on the Outbound Tasks link will display a red flag in the upper left corner until the Workflow is saved. Remember to Validate the Workflow as well to check all SQL Rules. Please see "To Validate a Workflow Map" on page 181 for more details.**

To add a Parallel Task Link

By deafult, Human Decision or Automatic Branch task types route projects along one of many outbound task links. However, a manager may wish other task types to route a document in two different directions within the company for dual work. To allow the Administrator to build multiple Outbound links for a task (other than Human Decision or Automatic Branch), the 'Allow Parallel Links' feature can be enabled.

◆ Under the Tasks menu, check/select 'Allow Parallel Links'.

NOTE: This enables Parallel Links for the duration of the current workflow designer session. A 'Consolidation' task type can be used to merge the different branches created by parallel links back into one.

To delete a link

1 **An Inbound link can only be deleted from the source task.**
You can click on the details window of the task with the Inbound link that you wish to delete, and double-click on the link itself. This will open the properties of the original, inbound task.

Then highlight the *Outbound* link on the original task.

Then we can either select *Designer|Delete|Workflow Task Link* from the menu, or use the *Delete Link* button on the tool bar to delete the link.

2 **To delete an Outbound link, select the link on the Outbound Link tab on the details window of the task that has the link. Either select *Designer|Delete|Workflow Task Link* from the menu, or click the *Delete link* button on the toolbar.**

The link is deleted.

Saving a Workflow Map

When making certain changes to a Workflow Map, you will be prompted to save by the system. For all other saving, you can select *Designer|Save* from the menu, or use the *Save* button on the tool bar to save. This will update the Documanage Server with your changes.

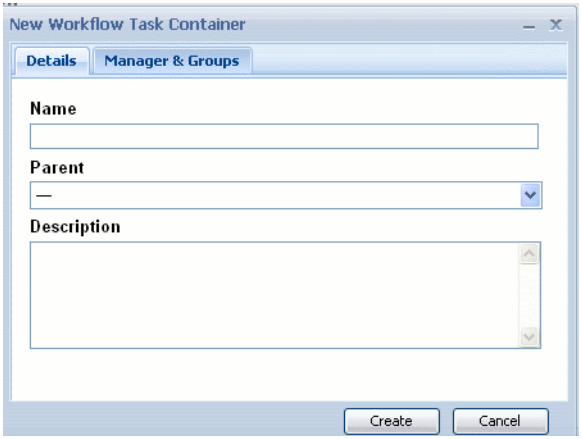
Working with Task Containers

Task Containers are optional, design time entities to assist Administrators in organizing workflow tasks. An Administrator can use them to group tasks, independent of their place in a workflow map flow design. These groups of tasks, in turn, can share a ‘parent container’, which will have default properties for task manager and team groups. A single task container could have tasks and other task containers as ‘children’. Task containers can be seen and utilized in the ‘Structure’ pane.

To add a new Task Container

1 **Select *Designer|New|Workflow Task Container* from the menu bar, or click on the *Create Workflow Task* button on the tool bar.**

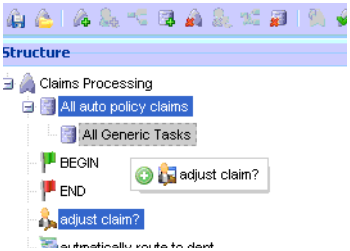
2 **The New Workflow Task Container dialog appears. The task container must have a name. If another parent container exists and this should be a sub-container of that one, select it from the drop-down list.**



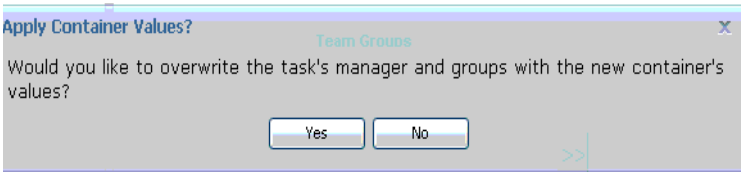
3 **The properties of the new Task Container display in the details pane. Note that changes to the properties of Task Containers will NOT**

change the properties of the Container's current child containers or child tasks. Only new tasks can reflect any new settings made here.

4 **To add a task to a Task Container, or to make an existing Task Container a child or another Task Container, click and drag the task or container onto the container you wish to add the task to.**



If the parent container has settings for Manager and/or Team Groups, you will see the following message:



To Delete a Task Container

1. **Select the Task Container to be deleted from the Structure pane. Use either *Designer|Delete|Workflow Task Container* from the menu, or use the *Delete Container* button on the tool bar.**

2. **The Delete Task Container confirmation dialog will appear.**

   Note that deleting a Container will not delete any child tasks or task containers from the Documanage System. These will be added back to the root of the Structure pane view.

Releasing a map

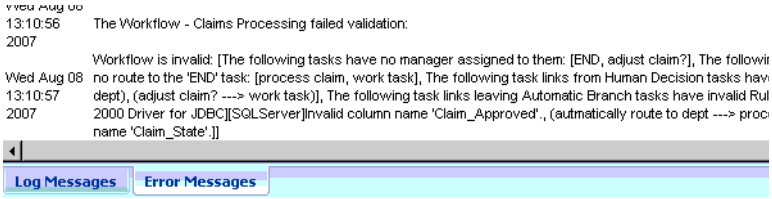
Only valid workflows can be released to workflow users for processing. A workflow is valid if all tasks are valid. Once the workflow is validated, it can be released. A workflow cannot be released until every task has a team and every task is linked from BEGIN to END. When a workflow is released, it is also activated. It is viewable to a user team once the server has been refreshed.

NOTE: Please see “To Validate a Task” on page 166 for instructions on validating a task.

If there are no users working in the workflow, it can be unreleased for alteration; if work is being done in the workflow, this could result in data loss. It is therefore recommended that a workflow be checked out if changes need to be made. Once the changes are complete, and the workflow passes validation, it can be checked in, and thus made available to users again. A workflow can also be deactivated so no further processing can be done on it by users.

To Validate a Workflow Map

1. **Select *Workflow Maps|Validate* from the menu, or click on the *Validate Workflow* button on the tool bar.**
2. **If successful, you will see the Workflow Validation Complete dialog.**
3. **If the workflow fails validation, you will see the Workflow Validation Failure message. When you click Ok on this message, you will see the reasons for the failure in the Error Message pane.**



To release a map

1. **Select *Workflow Maps|Release* from the menu.**
   The map is released. If a workflow is Released the word “Released” displays in the upper right corner of the workflow details window.

2. **The letter ‘R’ will appear next to the name of the workflow map on the Flow or Structure pane.**

3. **The workflow map will also be activated upon release. The word ‘Active’ will appear on the workflow details pane, and the letter ‘A’ will appear next to the map name on the Flow or Structure pane.**

Deactivating or deleting a map

If an Administrator wants to stop team members from accessing a map but does not want to delete it, he or she can deactivate the map. After it is deactivated, no further workflow projects may be started on the map. Existing projects on the map may continue to completion.

To deactivate a map

1. **Select *Workflow Maps|Deactivate*.**
2. **The word 'Active' will be greyed out in the workflow map's details window, and disappear from the name of the map in the Flow or Structure pane.**

To delete a map

1. **Select *Workflow Maps|Delete* or click on the *Delete Workflow Map*  button on the tool bar.**
2. **A confirmation message will appear. Click 'Yes' to confirm delete.**

NOTE: When a workflow is deleted, all references to it are removed.

To activate a map

1. **A workflow map that has been deactivated may be set to 'Active' again by selecting *Workflow Maps|Activate* from the menu bar.**
2. **The map should now be listed as 'Active' on the details pane, and have an 'A' next to the map name in the Flow or Structure pane.**

Editing a Released Workflow Map

A workflow that has been released and activated cannot be edited. If the Administrator needs to edit the workflow there are two options. One is to unrelease the workflow map. However, if there are users working in the map at that time, this could result in data loss, so must be used cautiously. The second option is to check out the workflow map, and then check it back in when changes are complete.

To unrelease a map

1. **Select *Workflow Maps|Unrelease* from the menu.**
2. **The details pane of the workflow map should now have 'Released' and 'Active' greyed out, and the workflow map name in the Flow or Structure pane will have no letters next to it.**

To check out a map

1. **To safely edit a workflow that may have users processing it, select *Workflow Maps|Checkout* from the menu or click on the *Checkout Workflow Map* button on the tool bar.**
2. **A 'C' will appear next to the map name in the Flow or Structure pane.**
3. **The workflow map will also have a number appended to it by the Documanage database while it is in the locked mode.**

   The original map will have a status of 'Locked' on the details pane, and the letter 'L' added to the 'R' and 'A' already present to show it is locked.

To Undo a Checkout

1. **You can Undo a Checkout by selecting *Workflow Maps|Undo Checkout* from the menu bar or clicking on the *Undo Checkout Workflow Map* button on the tool bar. This will delete the locked copy of the map, and will NOT apply any changes to the original map.**
2. **The 'Locked' status flag will disappear from the map details pane on the original workflow map, while the 'Released' and 'Active' flags**

will remain. The 'L' will also disappear from the name of the map as it displays in the Flow or Structure pane.

To Check In a Workflow Map

To save changes to a checked out workflow you must check the workflow map back in. This will unlock the original workflow and apply the changes, as well as delete the checked out copy from the Documanage System. If any task have been altered or deleted, this change will be applied safely, with no data loss. The users working in a deleted task would be able to complete the current project without any loss of work.

1. **To check-in a workflow map, either select *Workflow Maps|Checkin* from the menu, or click on the *Checkin Workflow Map* button on the tool bar.**

2. **If successful the original workflow will be opened, and the 'Locked' status will disappear from the details pane, and the 'L' will disappear from the name of the map in the Flow or Structure pane. The map will remain 'Released' and 'Active'.**

3. **If the checkin failed b/c the checked out copy of the map is invalid, the error messages will appear in the Error Message pane.**

Workflow Daemon

The Documanage Workflow Daemon automates workflow tasks that might normally be completed by human users. It can be run either as an NT service or as a console application. The daemon application logs into the Documanage Server with a user account, periodically checks the logged-in user's inbox for pending tasks and runs custom-programmed rules to process each task. The custom rules have access to the Documanage session and can

manipulate the workflow project and associated documents as desired, based on the privileges of the logged-in user account.

The custom programming and workflow set up required to make use of the daemon is beyond the scope of this document. Consequently, reference information for using daemon and custom programming rules to be used with it are described in the Documanage Programmers' Guide. The programmer should document each workflow function developed with the daemon. Set up information should be provided to the system administrator and workflow designer.

Graphical Workflow Designer

Introduction

NOTE: This feature is not available with the Documanage Server Lite product.

Documanage Workstation provides routing of documents along a pre-defined series of destinations (i.e., people or departments) in an organization. Documanage's Workflow Designer module enables you to design a workflow template which contains the steps needed to route documents for a particular project. Workflow Designer provides complete flexibility in handling workflows from the very basic to the very complex, depending upon your organization's needs.

In this chapter

The information in this chapter will help you answer the following questions:

- What are workflow templates?
- How do I use Workflow Designer to create workflow templates?
- How do I use workflow templates in projects?

Understanding workflow

A *workflow template* is a set of *tasks* and *links* which form the complete steps needed to route documents. A task is a destination where the documents are routed to or from. A link is the connection from one task to another. A template is processed by advancing from task to task along each link connecting them together.

Every workflow template is based upon data from a table in a database. The table that is used is called the ***workflow table.*** This table can be chosen from among all the tables that are mapped to Documanage. The documents in a workflow are associated with data within the workflow table (for example, in an auto parts workflow the documents in the workflow are associated with the data from the auto parts), to create workflow folders called *projects*. Documanage implements data driven workflows with the association between the data in the table and the documents. Note that you cannot use database views to create workflow templates.

A template is made available for processing by ***releasing it***. Once a template is released, it cannot be altered any further. A released template can be ***deactivated,*** or set so that no further processing can be done on it. A deactivated template can be reset for processing by ***activating*** it.

Workflow roles

There are several roles in the workflow model that are assumed by users.

- **Administrators**. Administrators are responsible for designing and maintaining a workflow template. An additional duty is monitoring a workflow for adequate throughput and lack of bottlenecks. All Administrators are database users within the DMANAGE database.

- **Managers**. Managers are responsible for assigning teams of people to various tasks. All managers are users whose names are taken from NT security. Only one manager is assigned to each task.

- **Team Members**. Team members are responsible for performing the various tasks in a workflow. Each team consists of one or more groups taken from NT security.

Getting started

The Graphical Workflow Designer is used to create the workflow template containing the steps needed to route documents for a particular project. The Workflow Designer allows you to:

- Create templates
- Create tasks and link them together
- Assign managers and team members to each task
- Release templates
- Activate and deactivate templates
- Delete templates.

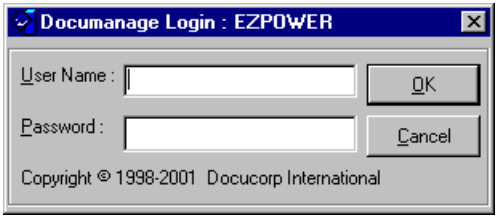
Before you can use Workflow Designer to create a workflow template, you must start and log in to Workflow Designer. Note that your start-up procedure may differ slightly from the one below.

To start the Graphical Workflow Designer

1 **Click and hold open the *Start* menu on the Windows Desktop.**

2 **Select *Programs|Documanage|Documanage Workflow Designer*.**

The *Login* dialog displays.



3 **Type your user name in the *User Name* text box, then press TAB.**

4 **Type your password in the *Password* text box, then click *OK*.**

The main Workflow Designer application window displays.

NOTE: The Graphical Workflow Designer validates the user name and password from database security. The user name must be that of a valid user of the DMANAGE database.

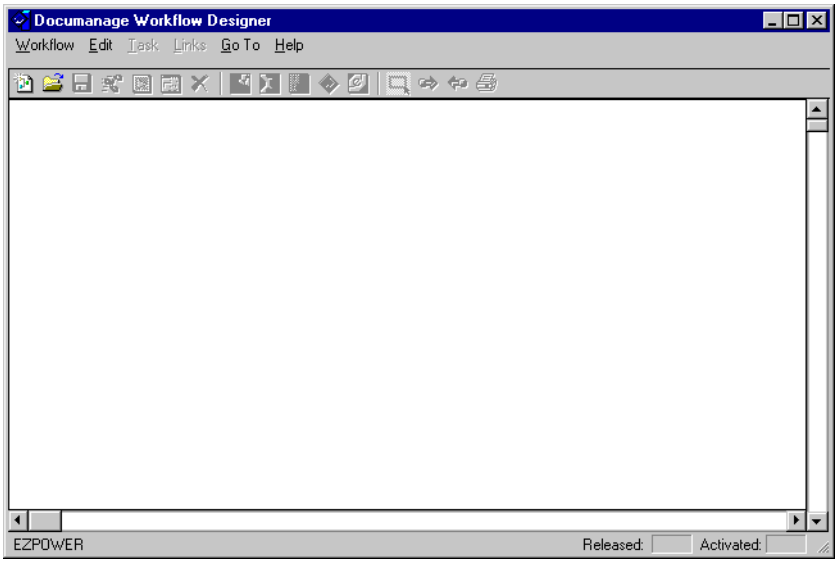
To exit Workflow Designer

1 **Choose *Exit* from the File menu.**

Graphical Workflow Designer quits the application and returns you to Windows with a confirmation message.

Using the main window

The Graphical Workflow Designer has several features that help you to create a workflow template.



- **Menu bar.** Menus provide access to commands that you use to create a workflow template. Some commands allow you to initiate actions immediately. Others display a dialog, which lets you specify file locations and settings and then initiate actions. Commands that are not available are dimmed.

  If the menu name has an underlined letter, press the *Alt* key and the underlined letter to move to the menu bar, select the menu, and open it. Once you've opened any menu, you can execute a command by typing the underlined letter of the command.

- **Tool bar.** The tool bar offers an alternative way to initiate standard commands. If a function is not available, the corresponding button on the tool bar is dimmed.

- **Drawing area.** The middle section of the Workflow Designer is the drawing window. All links and tasks are created in this window. If there is not enough room to add all the tasks and links, simply use the scrollbar to enlarge your viewing area.

- **Status bar.** The status bar at the bottom of the window indicates the status of the workflow. It lists the current user logged into Workflow Designer, and whether the template they are using is released and active.

- **Mode of operation.** When you are adding tasks, Workflow Designer enters *Add-Task mode*. When you are adding links, Workflow Designer enters *Add-TaskLink mode*. When you are selecting a task or link, Workflow Designer enters *Selection* mode. The mouse pointer changes depending on what mode you are in. Each mode only allows certain activities (for example you cannot add a task while in selection mode.)

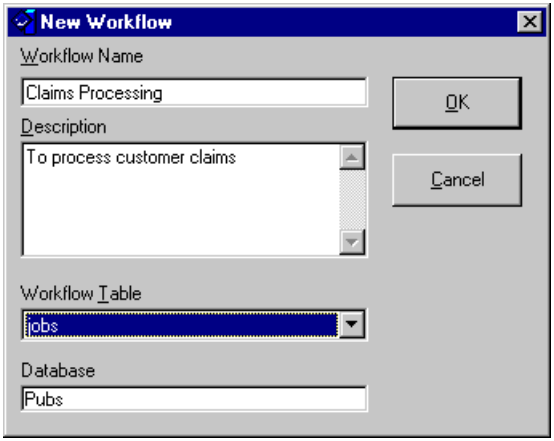
Creating a template

After you've started the Grahpical Workflow Designer and logged on, you are ready to create a workflow template. You can create a new template from scratch or by using the properties of an existing template (the properties may need to be altered or corrected after it is released.) All the links and tasks are kept intact, requiring minimal effort from a system administrator. This also serves to work-around the fact that changes cannot be made to a released template.

To create a new template

1 **Select *Workflow|New* or press [Alt+W+N].**

The *New Workflow* dialog displays.



2 **Type a template name in the *Workflow Name* text box, then press <Tab>.**

Enter a unique name up to 32 characters long. The workflow name should be descriptive of the process it describes. You can use spaces to improve readability. Duplicate template names and single quotes are not permitted.

3 **Type a description for the template in the *Description* text box, then press <Tab>.**

Enter a description up to 100 characters long that tells users who are unfamiliar with the workflow what its purpose is.

4 **Choose a table to base the template on from the *Workflow Table* drop-down box.**

This box includes all tables which are currently mapped to Documanage. When a table is chosen, the database in which it exists is automatically

displayed in the *Database* text box. The table you choose must contain the data that you want to associate with the documents.

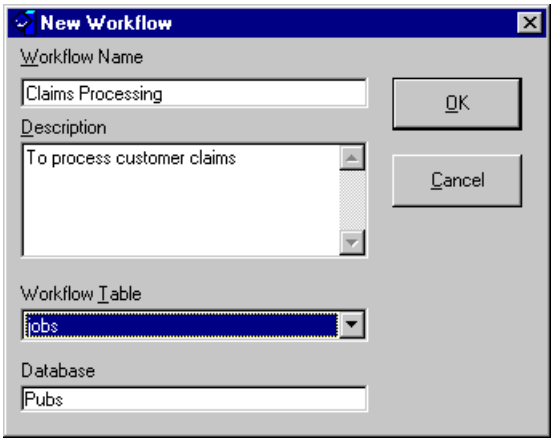
5 **Click *OK*.**

The *New Workflow* dialog closes, and the Login dialog displays.

6 **Type your user name in the User Name text box, then press TAB.**

7 **Type your password in the Password text box, then click OK.**

The workflow name displays in the Workflow Designer title bar, and the BEGIN and END tasks display in the Workflow Designer.



8 **To save your newly created workflow template, select *Workflow|Save*.**

If a template is not saved, the data is lost when exiting the Workflow Designer or when opening a different workflow.

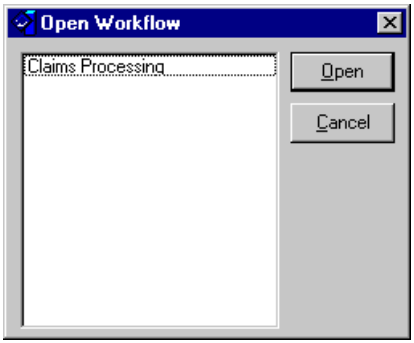
NOTE: A workflow template can also be saved as a bitmap file by selecting *Workflow|Save As Bitmap.* A dialog prompts you to specify a name for the new bitmap. Workflow Designer saves the currently open workflow template as a bitmap file in the specified location.

To open an existing template

The Graphical Workflow Designer lists most recently opened workflow templates at the bottom of the Workflow menu. This feature serves as a shortcut to open existing, recently accessed workflow templates. Simply select the workflow template's name from the Workflow menu, and it displays.

If the template you want is not listed on the Workflow menu, then follow the instructions below to locate and open an existing workflow template.

1 **Select *Workflow|Open*, or click the *Open* icon on the toolbar.**
The *Open Workflow* dialog displays.



2 **Double-click the workflow you want to open, or select and highlight the workflow and click *Open*.**

3 **To save the existing template as a new template, select *Workflow|Save As*.**

The *Save Workflow As* dialog displays.

4 **Type a new workflow name for the new template.**

When a template is open, tasks and links can be added, deleted, or changed as long as the template was not previously released.

You are now ready to add tasks to your newly created workflow template.

Working with tasks

The first step to perform after creating a workflow template is to add tasks. In order to add tasks properly, it is important to know what task types are available.

There are two system tasks required in a workflow template: “BEGIN” and “END.” There must be one and only one of each in a template. All the tasks of a template must be linked from the BEGIN task to the END task. The system automatically creates these tasks. They cannot be added to or deleted from a template.

- **BEGIN:** This task is the starting point of all templates. The team members assigned to this task are able to create projects.
- **END:** This task is the final task of a template. No team members can be assigned to this task.

In many instances, the route that a template takes is dependent upon the action taken in a previous task. These tasks are referred to as decision branches. There are two types of decision tasks from which you can choose: automatic branch tasks and human decision tasks.

- **Automatic branch task:** This task automatically routes a project to one of the tasks linked to it. It is routed to another task based upon the data in the workflow table. For example, Ford requires an automobile part supplier however all other car companies do not. *All parts with a make of Ford* are routed for approval while all other parts go to a different task. This step is automatically accomplished, precluding the need for human intervention.

  To use an automatic branch task, you must create a “rule” for every link from the automatic branch. These rules are based on the columns in the database table upon which the workflow is based. The links leaving an automatic branch task can have only a single DEFAULT rule. A workflow cannot be released if the Task Link properties of one or more of the links leaving its Automatic Branch tasks are set up incorrectly. A template advances to the link with the DEFAULT rule if the conditions set in any other rule are not met.

  An example of the automatic branch rules (using the auto parts example above) might be *make=’ford', and DEFAULT (for all other makes).*

NOTE: Creating proper rules for an automatic branch is important in order to ensure that the template is routed correctly. If you are unsure of how to create a rule, contact your Database Administrator for more details.

- **Human decision task:** This task gives the team member processing it the ability to route a project to one of the tasks linked to the human decision. When the task is processed, the team member chooses the appropriate

task to send the project to. For example, an auto parts documents would be sent to a project manager for review. That manager could then deny or approve the documents. The team member can not choose more than one task to route a project along.

- **Consolidation:** A template can be set up so that a task will not advance to the next task until all the tasks which are linked into it are processed. This type of task is a consolidation task. A template will not be advanced to a consolidation task until every parallel task is processed. For example, a manager may want to review all documents together for a part before giving final approval to the project. Until all documents are approved, the project will not be advanced.

Documanage also provides an Auto Mail task. An Auto Mail task is a special task which tells the Documanage Server to automatically send mail to a user while this task is processed. Unlike other tasks, this task will not show up in the user's Inbox.

Any tasks, which are not part of a category listed above, are considered generic tasks. Whenever an operation, such as adding a document to a file, needs to be performed a generic task can be used.

To add a task

1. **Click the toolbar icon representing the task type you want to add, or choose a task from the *Task* menu.**
2. **Move the mouse pointer to the position in the drawing area where you want to put the new task, then click the mouse button.**
3. **It cannot be on top of one which already exists. The icon representing the task displays on the template.**
4. **To reposition the task, click the *Selection* icon, click the task, and move the task while holding down the mouse button.**

To delete a task

1. **Click the *Selection* icon on the toolbar.**
   This puts the Workflow Designer in selection mode.
2. **Select the task to be deleted by clicking on it, then either select *Task|Delete* or click the *Delete* icon on the toolbar.**
   This deletes the task and all of its links.

To use Snap to Grid

Workflow tasks are automatically aligned on your dialog when you move them around if you've turned on Snap to Grid drawing.

- **Select *Edit|Snap to Grid* to toggle Snap to Grid mode on/off.**

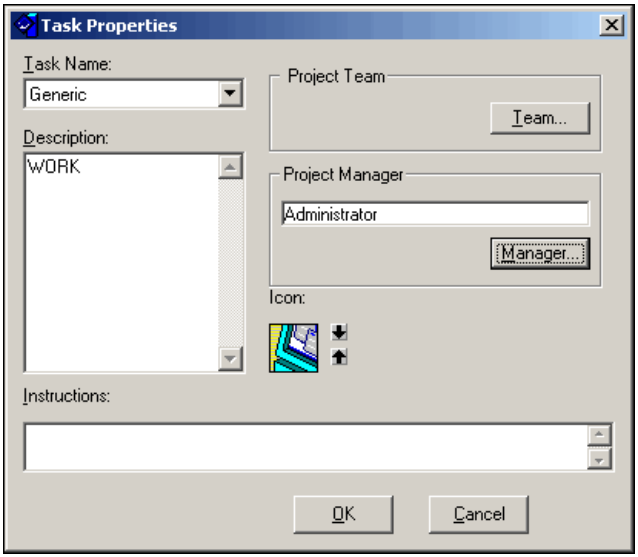
To specify a task's properties

It is necessary to specify task properties in order for a task to work properly within a workflow.

1. **Click the *Selection* icon on the toolbar.**
   This puts the Workflow Designer in selection mode.

2 **Select *Task|Properties*, double-click the task for which you want to set properties, or click the *Properties* icon on the toolbar.**

The *Task Properties* dialog corresponding to that task displays.



3 **To change the task name, click the *Task Name* drop-down box and choose the type of task with which you want to replace it.**

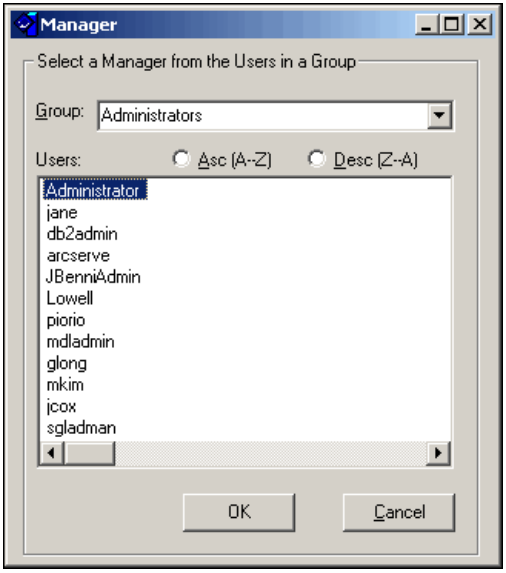
Every task must have a name, which is the type of task being used. This property can be changed, except in the case of BEGIN and END tasks.

4 **To change the description, click in the *Description* text box and change the text using the keyboard.**

Every task is required to have a description, which is what you see for the task when looking at the Workflow Designer window. It is also what a team member sees when they are processing the task. Each description should be unique, since this helps teams to process tasks. The default value for this field is always the task name except in the case of generic tasks when it is the word “WORK”.

5 **To change the manager, click *Manager.***

The *Manager* dialog appears.

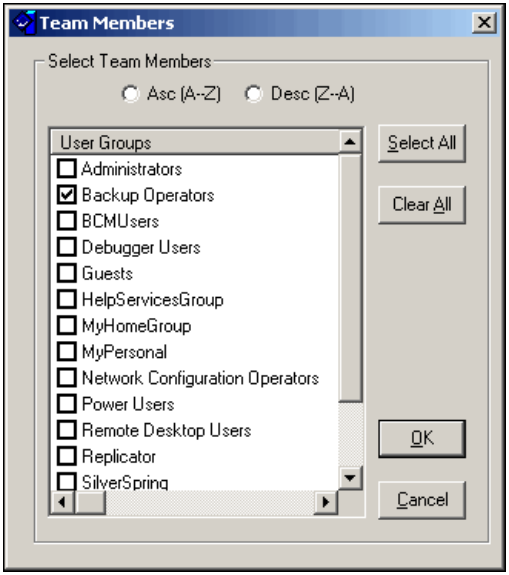


- From the list choose the user that you want to replace the present manager with.

NOTE: Every task must have a manager. The manager of the task is the person who is responsible for assembling each team to fill each role. A manager can manage more than one task. The team assigned to a role can be changed at any time.

6 **To add or remove a group assigned to a team's task, click the *Team* button.**

The *Team Members* dialog displays.



- To add a group to a team, select its check box, then click *OK*.
- To remove a group from a team, unselect its check box, then click OK.
- Click *Select All* to select all of the check boxes. *Select All* becomes unavailable.
- Click *Clear All* to clear all of the check boxes. *OK* becomes unavailable.

NOTE: Every task must have a team. The members of the groups on the team given to a task can process the task within Documanage Workstation.

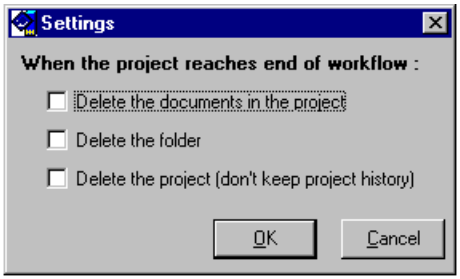
The groups are set up in Windows NT security. A task's team can have multiple groups assigned to it.

7 **To change the task icon, click the arrows next to the icon until you find the icon you want to use.**

A task must have an icon. You can change the icon of any generic task to a more appropriate one if you wish.

8 **If you selected an End task, click the *Project Settings* button on the *Task Properties* dialog.**

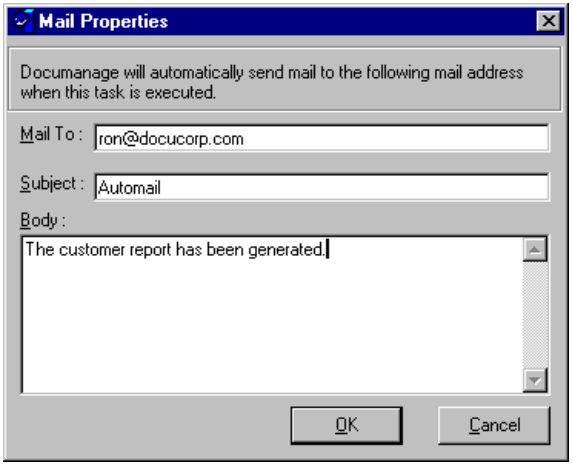
The *Settings* dialog displays.



- Check the appropriate option(s) if you want the documents, the folder, and/or the project that is added to the workflow to be deleted after it reaches the end of the workflow or to store it to provide project history.

9 **If you specified an Auto Mail task, click the *Mail Properties* button.**

The *Mail Properties* dialog displays.



An Auto Mail task is a special task that tells the Documanage Server to automatically send mail to a user while this task is processed. Unlike other tasks, this task will not show up in the user's Inbox.

NOTE: To use the Auto Mail task the server must be set up to mail.

Configure the mail settings for the task:

- In the *Mail To:* text box, type the mail address of the person to which this mail should be sent.
- In the *Subject* text box, type the subject of the mail.
- In the *Body* text box, type the mail text.
- Click *OK*.

10 **To change the instructions, click in the *Instructions* text box and change the text using the keyboard.**

This is an optional property. This property is used to help a team member when they are processing a task. If they need instructions on a task, they can click a *Help* button and display the instructions entered here.

Working with links

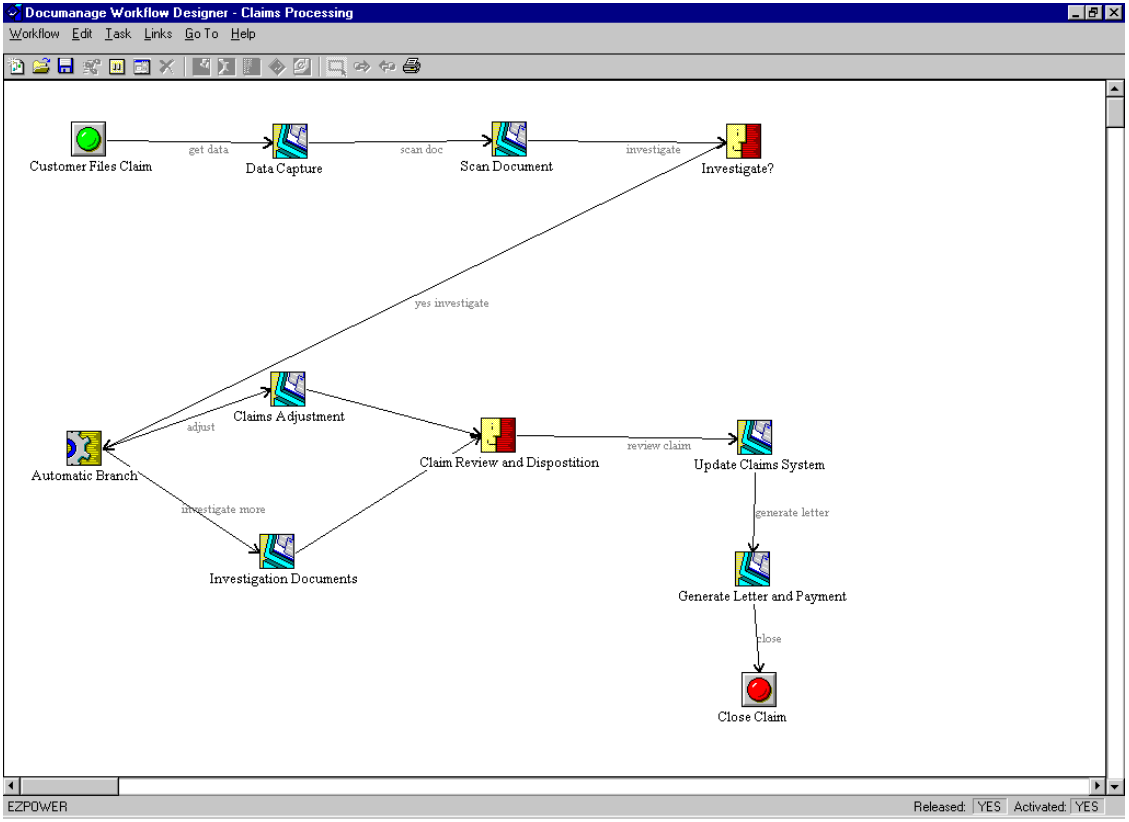
The next step to perform after creating a workflow template and adding tasks is to add “links.” Links connect one task within a template to another. Every task must be part of a link from the BEGIN task to the END task. When a task is processed, the project is routed to the tasks to which it is linked.

In order to add links properly, it is important to know what link types are available. There are two types of links used in Documanage: “forward” and “backward” links. The functionality of these links is the same. The reason that there are two types of links is to aid in the creation of a template. If the task being linked to is to the left of the linking task then a forward link would appear to be confusing. In these cases, a backward link should be used.

To add a link

1. **Click the *Forward link* or *Reverse link* icons on the toolbar.**
2. **Click the task which will be linked from.**
3. **While holding the mouse button down, drag the mouse pointer to the task which will be linked to.**
4. **Release the mouse button.**

   An arrow displays pointing from the first task to the second. This arrow is the link.



To delete a link

1. **Click the *Selection* icon on the toolbar.**
   This puts the Workflow Designer in selection mode.

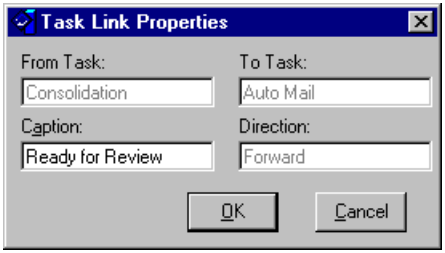
2. **Select the link to be deleted by clicking on it, then either select *Links|Delete* or click the *Delete* icon on the toolbar.**
   The link is deleted.

To specify a link's properties

1. **Click the *Selection* icon on the toolbar (this puts the Workflow Designer in selection mode), then click the link.**

2. **Either double-click the link, or click the *Properties* icon on the toolbar, or select *Links|Properties*.**

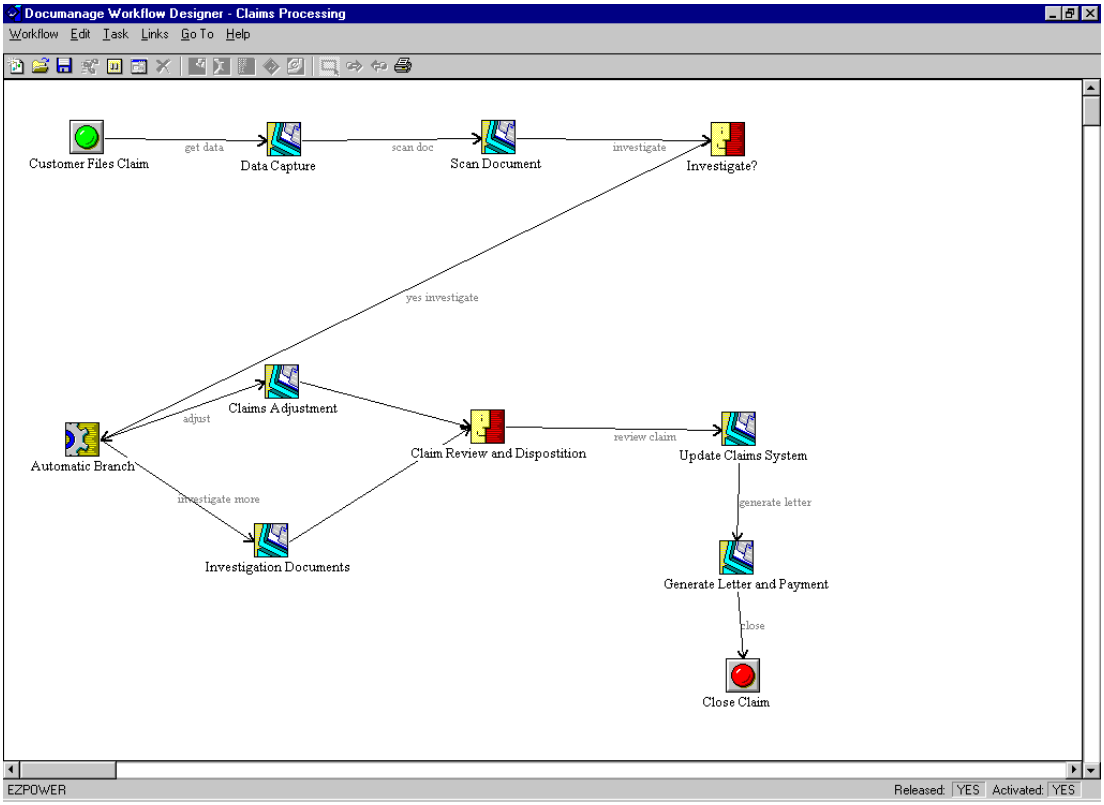
   The *Task Link Properties* dialog displays.



Some link properties are editable by the user and some are not. The different link properties are listed below.

- **From Task property of a link:** This property is not editable. This property indicates the task where the link originates.
- **To Task property of a link:** This property is not editable. This property indicates the task the link goes to.
- **Direction property of a link:** This property is not editable. This property indicates whether it is a forward or reverse link.
- **Caption property of a link:** The caption property is editable by the user. The caption of a link serves two purposes. First, when creating or editing a template, captions help to keep track of the purpose for each task. Second, when the task being linked from is a human decision, the caption for the link tells users what decision choices they can make. For example, the dialog below shows a sample Auto Parts Documentation template with link captions of "Approve" and

"Deny" from the two human decision tasks (Approval of Artwork and Approval of tech docs.



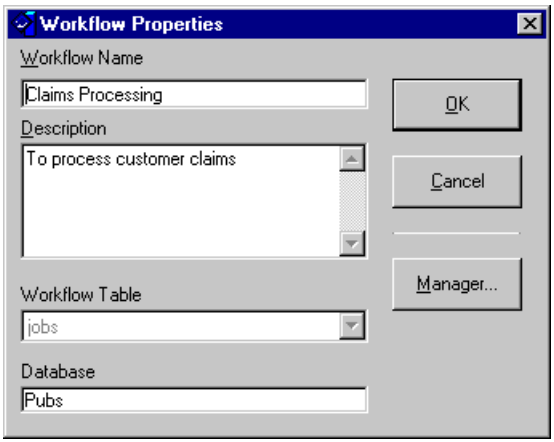
Changing workflow properties

You can change the workflow manager, description, and workflow table for a workflow template at any time.

To change workflow properties

1 **Select *Workflow|Properties*, or click the *Properties* icon on the toolbar when you are not focused on a link or task.**

The *Workflow Properties* dialog displays.



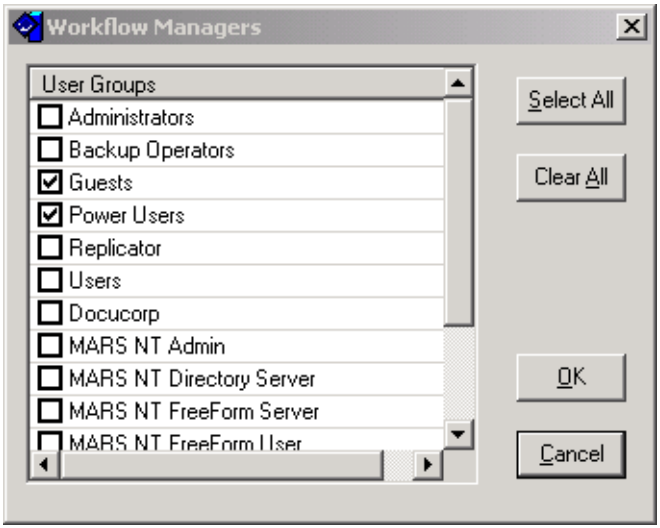
2 **To change the description, type a new description in the *Description* text box.**

3 **To change the table, click the *Workflow Table* drop-down box and choose a new table.**

After you have chosen a table, the database indicated in the database textbox changes automatically if the databases are different.

4 **To change the manager, click the *Manager* button.**

The *Workflow Managers* dialog appears.



- To add a group that the manager belongs to, select its check box, then click *OK*.
- To remove a group, unselect its check box, then click *OK*.
- Click *Select All* to select all of the check boxes. *Select All* becomes unavailable.
- Click *Clear* to clear all of the check boxes. *OK* and *Clear* become unavailable.

Releasing a template

To make a workflow template available for team members to use, it must be released. When a workflow is released, no further alterations can be made to it. It is viewable to a team once the server has been refreshed. A workflow cannot be released until every task has a team and every task is linked from BEGIN to END.

To release a template

1. **Select *Workflow|Release*, or click the release icon on the toolbar.**
   The template is released. If a workflow is Released the word “YES” displays next to the word “Released” and “Activated” at the bottom of the dialog.

2. **Login to the database containing the table the workflow was based upon.**
   Type a User ID that has the right to create views within the database.

3. **Click *OK*.**
   The word “YES” displays next to the words “Released” and “Activated” at the bottom right side of the dialog.

4. **Select *Workflow|Refresh Server* to refresh the server.**
   In order for a team to see a released workflow, the server must be refreshed. Once the server is refreshed, the workflow can be used. Any subsequent time the server is started, the workflow is usable.

Deactivating or deleting a template

If an Administrator wants to stop team members from accessing a template but does not want to delete it, he or she can deactivate the template. After it is deactivated, no further workflow projects may be started on the template. Existing projects on the template may continue to completion.

To deactivate a template

1. **Select *Workflow|Deactivate*.**
2. **Select *Workflow|Save***
3. **Select *Workflow|Refresh* Server.**

   The template is deactivated. If a workflow is deactivated the word “NO” displays next to the word “Activated” at the bottom of the dialog.

To delete a template

1. **Select *Workflow|Delete*.**

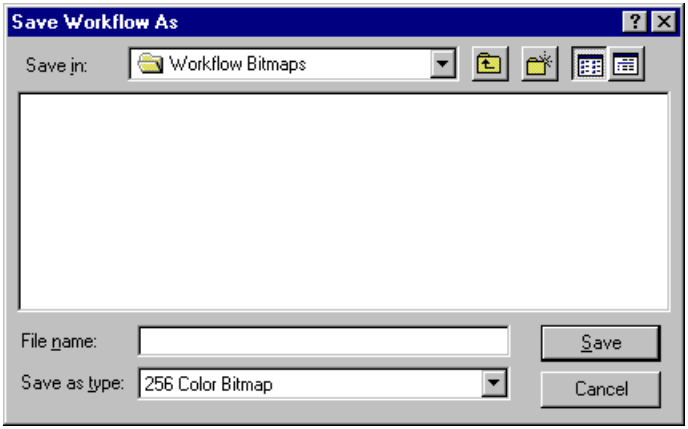
   A login dialog displays
2. **Login into the database**
3. **Type a User Name and Password**
4. **Select Workflow|Refresh Server to refresh the server.**

NOTE: When a workflow is deleted, all references to it are removed.

Saving a template as a bitmap

You can save the currently open workflow template as a color bitmapped image for printing and use in documentation and review.

1 **Select *Workflow|Save As a Bitmap*.**
The *Save Workflow As* dialog displays.



2 **Type a name for the image in the File Name text box, then click *Save*.**
The workflow is saved as a bitmap in the designated location.

Workflow Daemon

The Documanage Workflow Daemon automates workflow tasks that might normally be completed by human users. It can be run either as an NT service or as a console application. The daemon application logs into the

Documanage Server with a user account, periodically checks the logged-in user's inbox for pending tasks and runs custom-programmed rules to process each task. The custom rules have access to the Documanage session and can manipulate the workflow project and associated documents as desired, based on the privileges of the logged-in user account.

The custom programming and workflow set up required to make use of the daemon is beyond the scope of this document. Consequently, reference information for using daemon and custom programming rules to be used with it are described in the Documanage Programmers' Guide. The programmer should document each workflow function developed with the daemon. Set up information should be provided to the system administrator and workflow designer.

Appendix A: Managing Heterogeneous Servers in Documanage

A Documanage system consists of a Documanage Router and one or more Documanage Servers. The router may run on Windows NT/2000/XP, or on one of several UNIX-based operating systems, including Linux, Solaris, and AIX. A server may run on any of these same operating systems, and a single router may host servers based on more than one operating system. For example, a Documanage installation may include an NT-based router, an NT-based server, and a Linux-based server.

In Documanage the presence of multiple servers is transparent to the user. The user connects to the router and is assigned a server. The user does not know which server she is using, and could be connected to a different server each time she logs in. Each server will provide access to the same database and to the same document data store.

In order for each server to provide access to the same data store, the store must be accessible to each server through a network connection. The OT_Volumes table in the database contains the Documanage-specific name of the storage location, and an encrypted string containing the path or storage specification. This string can be edited (in plain text) in the Storage dialog of the Documanage Administrator application.

BLOB data is generally kept in the same database as the document data, and is available to each server through its database connection. VLAM data is stored on a mainframe and is accessible to each server through a set of code libraries.

Standard file system storage is accessed locally or through a network share, such as a shared volume on an NT server, or a Samba or NFS shared volume on a Linux server.

When a data store is specified in the Documanage Administrator, a standard notation is required. For example, BLOB storage is specified as BLOB|DSNAME\OWNER.BLOBTABLE. This notation is recognized by servers running on any platform.

Shared storage presents a special problem in that mounted volumes and directories are specified differently on different platforms. For example, a server on Windows 2000 may use a networked storage volume, using the Universal Naming Convention (UNC)-formatted path \\SomeMachine\DMGStorage. Prior to the introduction of UNIX-based Documanage servers, this is how file system storage was specified in the Administrator. However, a server on Linux would address this directory differently and would not recognize the Windows directory specification. A Linux server might expect a path such as /usr/DMGStorage, where /usr/DMGStorage is the mount point for the network share.

In a homogeneous system in which all the servers are running on the same platform, this is not an issue. In a system that supports only Windows servers, the native or UNC path is used. In a system that supports only Unix servers, the native path is used. In a heterogeneous system that supports a mix of Windows and Unix servers, the UNC path is used. On the Linux machines, the volume must be mounted to a mount point that maps to the UNC path by replacing "\" characters with "/" characters.

If you are configuring a system that uses all Windows servers, simply enter the UNC path to the storage in the Storage dialog of the Documanage Administrator. For example, you might enter "Default" in the Volume field, and "\\SomeMachine\DMGStorage" in the Path field.

If you are configuring a system that uses all Linux or Unix servers, you must mount the shared storage volume to the same mount point on each server machine. Then enter the path to that mount point in the Storage dialog. For example, you might enter "Default" into the Volume field, and "/usr/DMGStorage" into the Path field.

If you have both Windows and Unix/Linux servers in your system, determine the UNC path that the Windows servers will use to find the storage. Enter it into the Storage dialog of the Administrator. On the Linux machines, create the necessary directories and mount the shared storage to a mount point that can be expressed in the same path, with "/" replacing "\". For example, mount \\SomeMachine\DGMStorage to /SomeMachine/DMGStorage. If you need to mount further down in the file system hierarchy, you can enter a preface in the PowerOfficeServer section of the poffice.ini file. Enter "StorageRoot=[preface]". For example, if you want to mount the above volume at /usr/SomeMachine/DMGStorage, enter "StorageRoot=/usr". The default storage root is a blank string, so if you want to mount at /SomeMachine/DMGStorage, no StorageRoot entry is necessary.

Here are some examples of how various paths map to Windows and UNIX.

Storage Root	Configured String in Database	Path on Windows	Path on Unix
""	\\PC\DMGStorage	\\PC\DMGStorage	/PC/DMGStorage
"/usr”	\\PC\DMGStorage	\\PC\DMGStorage	/usr/PC/DMGStorage
<ignored>	D:\DocStore\Vol1	D:\DocStore\Vol1	<NA>
<ignored>	/usr/DMGStorage	<NA>	/usr/DMGStorage
<ignored>	BLOB|DS\Owner\ Table	BLOB|DS\Owner\ Table	BLOB|DS\Owner\Table
<ignored>	VLAM|LibName	VLAM|LibName	VLAM|LibName

Appendix B: Documanage .ini File Settings

File sections

The following is a list of the sections in the poffice.ini file. Each entry in the list contains a short description, a name/value table, and a sample file entry.

PowerOfficeRouter

The PowerOfficeRouter section contains the configuration entries for the Documanage router. This must include its RPC settings.

Name	Possible Values	Default	Description
LoginGroup	Valid group name	"Users"	Defines the group that all users must belong to in order to log in to the Documanage system.
NetworkAddress	Valid IP address	No Default	Defines the network address the Documanage Router uses to configure its RPC service.Note: This value is referenced by the Documanage Workstation.

Name	Possible Values	Default	Description
EndPoint	Valid endpoint value based on the protocol	No Default	Defines the protocol specific endpoint (for example, the TCP/IP port number) the Documanage Router uses to configure its RPC service.Note: This value is referenced by the Documanage Workstation.
Protocol	One of the possible RPC protocol descriptors.	"ncacn_np"	Defines the communication's protocol used by the Documanage Router RPC endpoint.Note: This value is referenced by the Documanage Workstation.
Options	RPC binding options	No Default	Binding options string supplied to "RpcStringBindingCompose()" call when router connects to server.
Domain	Valid network domain ID.	No Default	Defines the default domain used by the Documanage Router for authentication and authorization purposes.
RouterName	Any appropriate string	"/.:/DocumanageRouter"	Names the instance of the router
LogDetailLevel	0 = low1 = medium 2 = high/extended	0	Specifies the detail level for log.
LogExtraDetail	0 = Do Not Add Detail 1 = Add Detail	0	Specifies whether to log extra information with each entry or not. This information includes the module name and calling function.
LogDisable	0 = Allow Logging 1 = Disable Logging	1	Specifies whether to allow logging or disable it.
LogFile	Valid file path	"dmgrouter.log"	Defines the name of the Documanage Router logfile.
LogMinSize	Size in Kbytes	100	DEPRECATED
LogMaxSize	Size in Kbytes	200	Specifies the maximum amount of data to be added to the current log file before archiving.

Name	Possible Values	Default	Description
LogSqlLevel	0 = Off 1 = Non-Select State-ments 2 = All Statements	0	Specifies the detail level for SQL log entries.
DeleteFileAtStartup	0 = Do Not Delete 1 = Delete	0	Specifies whether to delete the current log file upon server startup of to append new entries.
LoadReportingTime-out	2 | 3 | 4 | …	No Default	DEPRECATED

Sample ini file entry:

```
[PowerOfficeRouter]
NetworkAddress=127.0.01
Protocol=ncacn_ip_tcp
Endpoint=4000
Domain=MDLDOMAIN
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
LogFile=DMGRouter.log
LogMinSize=100
LogMaxSize=200
LogSqlLevel=2
DeleteFileAtStartUp=0
```

PowerOfficeServer

This section defined the configuration for the Documanage Server application. This may include items such as its RPC settings and debug/trace settings.

Name	Possible Values	Default	Description
NetworkAddress	Valid IP address	No Default	Defines the network address the Documanage Server uses to configure its RPC service. This is optional.
EndPoint	Valid endpoint value based on the protocol	No Default	Defines the protocol specific endpoint (for example, the TCP/IP port number) the Documanage Server uses to configure its RPC service. This is optional.
Protocol	One of the possible RPC protocol descriptors.	"ncacn_np"	Defines the communication's protocol used by the Documanage Server RPC endpoint. This is optional.
Options	RPC binding options	No Default	Binding options string supplied to "RpcStringBindingCompose()" call when server connects to router.
LoadReport-WeightFactor	Any positive int	1	Changes the weight of a server during router load balancing.
IndentifyEmpty-Folder	0 | 1	0	Specifies whether or not to return empty folders.
KeepHistory	0 | 1	1	Specifies whether or not to retain the workflow's activity log history.
EnableNewTasks	0 | 1	1	Specifies whether new tasks are Auto Mail, Launcher and Poller enabled
EnableAutoma-tionForPoller	0 | 1	1	Specifies whether we are using an automation dll or a regular dll for POLLING

Name	Possible Values	Default	Description
MAPIProfile	Any valid MAPI profile name	Microsoft Outlook Internet Settings	Specifies the MAPI profile to be used by the Auto Mail task.
RenderTemp	A valid directory path fragment	'temp' subdirectory of the server directory	Specify the temp directory for remoterex, page at a time rendering
RenderLog	A valid log file name	"RemoteRex.log"	Specify the log file for remoterex, page at a time rendering
RenderLogLevel	0 = off 4 = warning 5 = error else information	0	Specify the log level for remoterex, page at a time rendering
RenderINSO	A valid directory path	'inso' subdirectory of the server directory	Specifies the directory containing the INSO imaging libraries used for some file types.
EZPOWERDatabase		No Default	DEPRECATED
AutoLogin		No Default	DEPRECATED
LogDetailLevel	0 = low 1 = medium 2 = high/extended	0	Specifies the detail level for log.
LogExtraDetail	0 = Do Not Add Detail 1 = Add Detail	0	Specifies whether to log extra information with each entry or not. This information includes the module name and calling function.
LogDisable	0 = Allow Logging 1 = Disable Logging	1	Specifies whether to allow logging or disable it.
LogFile	Valid file path	poffice.log	Defines the name of the Documanage Server logfile.
LogMinSize	Size in Kbytes	100	DEPRECATED
LogMaxSize	Size in Kbytes	200	Specifies the maximum amount of data to be added to the current log file before archiving.

Name	Possible Values	Default	Description
LogDebugTrace	0 = No 1 = Yes	0	Specifies whether logging is also sent to the debugger output
LogSqlLevel	0 = Off 1 = Non-Select State-ments 2 = All Statements	0	Specifies the detail level for SQL log entries.
LogSqlDriver-Calls	enter | exit | all | exec | fetch | bind | con-nection | resources	“ “	'enter' and 'exit' determine whether driver calls will be logged before the call, after the call, or both. 'exit' log-ging also logs the time spent in the call, in milliseconds. The following keywords determine which driver calls to log; one or more may be set. - all: log all driver calls - exec: log SQLExecute and SQLExec-Direct calls - fetch: log SQLFetch calls - bind: log SQLBindCol and SQLBind-Parameter calls - connection: log calls related to con-necting, disconnecting, and setting connection options. Includes transac-tions. - resource: log calls that allocate or free resources, such as statement handles. - schema: log calls that look at the database catalog or otherwise deter-mine the database schema informa-tion.

Name	Possible Values	Default	Description
LogAutoSave	0 = Buffer the log file and write several lines to disk at a time 1 = Close the log file after every write	1	When set to 1, the server writes to the log file and closes the file after every line is written. Setting it to 0 makes logging faster, but if the server crashes you may miss the last few entries, and if you open the file while the server is running, you will be unable to write to it.
DeleteFileAtStar-tup	0 = Do Not Delete 1 = Delete	0	Specifies whether to delete the current log file upon server startup of to append new entries.
ShowTree	0 | 1	1	DEPRECATED
ShowUsers	0 | 1	1	DEPRECATED
ShowCalls	0 | 1	1	DEPRECATED
GarbageCollec-tionDelay	Delay in seconds	10	DEPRECATED
FillVolumesSe-quentially	0 | 1	0	DEPRECATED
MaxConcurrent-Calls	Any Integer	20	Maximum number of concurrent client calls to the server. Not limit for sessions, but actual in-progress concurrent operation calls.

Sample ini file entry:

```
[PowerOfficeServer]
NetworkAddress=127.0.0.1
Protocol=ncacn_ip_tcp
Endpoint=4100
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
LogFile=poffice.log
```

```
LogMinSize=100
LogMaxSize=200
LogSqlLevel=2
LogSQLDriverCalls=
DeleteFileAtStartUp=0
LogAutoSave=0
```

Sample LogSQLDriverCalls entries:

```
; see how long each call to execute a statement or fetch
  data takes:
      LogSQLDriverCalls=exit exec fetch
; see how and when connections are made and statements
  are allocated:
      LogSQLDriverCalls=enter connection resource
; maximum logging:
      LogSQLDriverCalls=enter exit all
; no SQL driver logging:
      LogSQLDriverCalls=
```

EZPDSInfo

The EZPDSInfo section defines the Documanage data source used by the router and server applications.

Name	Possible Values	Default	Description
DataSource	Valid DSN	"EZPOWER"	Defines the Data Source Name (DSN) for the Documanage database.
AuthID	Valid DSN Owner String	"EZPOWER"	Defines the table owner value for the Documanage database. NOTE: If AuthID is not a valid string then the Configure DSN section of the Documanage Administrator generates errors and no sections of the Administrator can be completed.

Sample ini file entry:

```
[EZPDSInfo]
DataSource=DMGSAMPLES
AuthID=dbo
```

Directories

The Directories section defines the paths for directory resources used by the router and server applications.

Name	Possible Values	Default	Description
Pictures	Valid Directory Path	"C:\POFFICE\PICTURES"	Defines the directory where <some pictures?> are stored/accessed.

Sample ini file entry:

```
[Directories]
Pictures=.\Bitmaps\
```

StorageProviders

The StorageProviders section defines the plug-in shared libraries that provide access to the document storage mechanism (filesystem, database, etc).

Name	Possible Values	Default	Description
NTFS	Plug-in Filename	No Default	Provides the name for the NTFS storage plug-in.
BLOB	Plug-in Filename	No Default	Provides the name for the database storage plug-in.
VLAM	Plug-in Filename: The name of the ini file used by CommCommander	No Default	Provides the name for the VLAM storage plug-in. There are two versions of the plug-in: dmgvlam20.dll and dmgvlam21.dll. The settings are the same for both plug-ins, but they interact with versions 2.0 and 2.1 of CommCommander, respectively.
SNAP	Plug-in Filename	No Default	Provides the name for the SNAP storage plug-in
EMCC	Plug-in Filename	No Default	Provides the name for the EMCC storage plug-in

Sample ini file entry:

```
[StorageProviders]
NTFS=dmgntfs.dll
BLOB=dmgblob.dll
VLAM=dmgvlam21.dll:isi.ini
SNAP=dmgsnap.dll
EMCC=dmgemcc.dll
```

AuthorizationProviders

The AuthorizationProviders section defines the available authorization mechanisms for user validation (user group memberships). These are also plug-in shared libraries.

Name	Possible Values	Default	Description
NTLM	Plug-in Filename	No Default	Specifies the authorization provider plug-in for NTLM
DMGDB	Plug-in Filename	No Default	Specifies the authorization provider plug-in for DMGDB
LDAP	Plug-in Filename	No Default	Specifies the authorization provider plug-in for LDAP

Sample ini file entry:

```
[AuthorizationProviders]
DMGDB=DMGDB_Authorization.dll
```

AuthenticationProviders

The Authentication Providers section defines the available authentication mechanisms for user validation (user name and password check). These are also plug-in shared libraries.

Name	Possible Values	Default	Description
Default	Plug-in Filename	No Default	Specifies the default authentication provider plug-in. This overrides the OS specific default plug-in (Windows = NTLM, IX's = DMGDB).
<UserClass>	Plug-in Filename	No Default	Currently NOT in use and may change. Specifies the authentication provider plug-in to use to authenticate users belonging to the specifies class (NYSales=NTLM_AuthenticationServer.dll means that all users in the NYSales class will be authenticated using NTLM).

Sample ini file entry:

```
[AuthenticationProviders]
Default=DMGDB_AuthenticationServer.dll
```

LDAPAuthoritiesConfiguration

The Lightweight Directory Access Protocol, or LDAP, queries and modifies directory services running over a TCP/IP network. The LDAP plugin supplied with Documanage interfaces a Documanage server to an LDAP server. This allows the Documanage server to use data from an LDAP server to authenticate and authorize Documanage users. Settings stored in the LDAPAuthConfig section of the poffice.ini file configure the plugin.

Activating the Plugin

To activate the plugin so that it reads the entries from this section of the poffice.ini file, you must make the settings in other sections of the file that are listed here:

Section	Entry	Value	Description
PowerOfficeRouter	Login-Group	dmgusers	Tells the Documanage the common name of the group entry that identifies users
AuthenticationProviders	Default	ldap_authenticationserver.dll	Configures the Documanage server for LDAP authentication
AuthorizationProviders	LDAP	ldap_authorization.dll	Configures the Documanage server for LDAP authorization

Sample ini file entry:

```
[PowerOfficeRouter]
NetworkAddress=127.0.01
. . .
LoginGroup=dmguser
[AuthenticationProviders]
Default=ldap_authenticationserver.dll
```

```
[AuthorizationProviders]
LDAP=ldap_authorization.dll
```

LDAPAuthConfig Entries

The entries in the `LDAPAuthConfig` section of the poffice.ini file are listed and defined here:

Name	Possible Values	Default	Description
Configuration for LDAP server connection:			
Host	**10.7.10.100 or mybox.mydo-main**	**localhost**	IP address or name of the host machine for the LDAP server.
Port	**nnnn**	**389**	The TCP port used by the LDAP server running on the host.
Binddn	**yourID**	**anonymous**	Distinguished Name (DN) for the base name context; used for logging into the LDAP server.
Bindpw	**yourPW**	**“ “**	Password (PW) for logging into the Documanage account on the LDAP server.
General configuration for searches:			
Base	**dc=example, dc=com**	**“ “**	Base object for the LDAP searches
Scope	**sub, top, one**	**sub**	Scope of the LDAP searches
Configuration for user entries:			
Uidattr	**uid**	**uid**	User ID attribute
Uidpw	**pw, MD4**	**pw, clear**	Identifies the password attribute and its representation.
Uidfriendlyattr	**CN**	**cn**	The attribute name in which a full name for the user appears.

Name	Possible Values	Default	Description
Uidgroupattr	**memberOF**	**“ “**	Establishes group membership in the user entry. An empty string indicates that there is no way to indicate group membership in a user entry.
Uidfilter	**(&(ou=docu-corpUs-ers)(objectCla ss=docucor-pAccount))**	**(&(ou=docucorpUs-ers)(objectClass=docu-corpAccount))**	A search expression, or filter, for finding user entries. Search expressions support AND and OR prefix operators and wildcards. (objectClass=*) returns all of the entries in an LDAP server.
Uidgroup-memberclass	**yourGroupOU**	**“ “**	The organizational unit for group entries.

Name	Possible Values	Default	Description
Configuration for group entries:			
Groupfilter	**(&(ou=docu-corp-Groups)(objectClass=group of Unique-Names))**	**(&(ou=docucorp-Groups)(object-Class=group of UniqueNames))**	A search expression, or filter, for finding group entries. Search expressions support AND and OR prefix operators and wildcards. (objectClass=*) returns all of the entries in an LDAP server.
Groupnameattr	**cn**	**cn**	The specific short name of a group. It identifies the attribute name for membership in a group.
Groupmember-attr	**uniqueMem-ber**	**uniqueMember**	The attribute name for membership in a group
Groupfriend-lynameattr	**cn**	**description**	Description; the attribute name of a long name that describes the group
Groupuidmem-berclass	**yourUserOU**	**docucorpUsers**	Identifies the objectClass for the user members of a group. The organizational unit for user entries.

Sample ini file entry:

```
[LDAPAuthConfig]
; config for LDAP server connection
Host=10.7.10.163
Port=389
Binddn=
Bindpw=

; general config for searches
Base=dc=example,dc=com
Scope=sub
```

```
; config for user entries
Uidattr=uid
Uidpw=sambaNTPassword,MD4
Uidfriendlyattr=cn
;Uidgroupattr=memberof
Uidfilter=(objectClass=docucorpAccount)
Uidgroupmemberclass=docucorpGroups


; config for group entries
Groupfilter=(objectClass=groupofUniqueNames)
Groupnameattr=cn
Groupmemberattr=uniqueMember
Groupfriendlynameattr=description
Groupuidmemberclass=docucorpUsers
```

PowerOfficeDesigner

The PowerOffice Designer section defines the runtime characteristics for the Documanage Workflow Designer.

Name	Possible Values	Default	Description
EnableNewTasks	1 = Enable 0 = Disable	0	Specifies whether the Workflow Designer will allow users to create new workflows or not.
RecentWorkflowCount	0 or any positive integer	0	The number of workflows in the Documanage system.
Item<#>	Workflow name	No Default	The name of a recent workflow accessed by the designer

Sample ini file entry:

```
[PowerOfficeDesigner]
EnableNewTasks=1
RecentWorkflowCount=1
Item1=Claim Properties
```

PowerOfficeClient

The PowerOfficeClient section defines the runtime characteristics for the Documanage workstation.

Name	Possible Values	Default	Description
MaxFolders	Positive Integer	No Default	DEPRECATED.
RequestedProtocol		No Default	DEPRECATED
ControlErrorLogging		No Default	DEPRECATED
MaxCabinets		No Default	DEPRECATED
ViewDir	Valid directory path	Windows System32 directory	DEPRECATED
Options	RPC binding options	No Default	DEPRECATED.
TimeOut	Positive Integer	Zero (0) (uses server setting)	DEPRECATED.

Sample ini file entry:

```
[PowerOfficeClient]
MaxFolders=100
```

Cabinet Filters

The Cabinet Filters section contains picklist values for cabinet filter variables. Refer to the description of the “Variable” field on page 13. When you open a cabinet for which a picklist for a filter is defined, you will be presented with a list of values for its filter variables.

Name	Value	Description
Variable Name	number of picklist items	The number of picklist items defined for the variable name
Variable Name0	value for first picklist item	The picklist variables start with zero.
Varialbe Name1	value for the second picklist item	This is the second item since the picklist variables start with zero.

Sample ini file entry:

```
[Cabinet Filters]
State_Code=5
State_Code0=MD
State_Code1=GA
State_Code2=TX
State_Code3=NC
State_Code4=CA
```

DesktopImport

The DesktopImport section defines the desktop folder into which documents may be placed for importation into the Documanage system (when the workstation is next run).

Name	Possible Values	Default	Description
WatchDirectory	Valid Directory Path	C:\Documents and Settings\[Current User]\Desktop\Import to Documanage	DEPRECATED.These options are now maintained by the Documanage Workstation Options dialog.
PollingRateInMilliSeconds	Wait Timeout	65535	DEPRECATED.These options are now maintained by the Documanage Workstation Options dialog.
UsePoller	1 = Poll 0 = Don't Poll	1	DEPRECATED.These options are now maintained by the Documanage Workstation Options dialog.
ShowMessage	True | False	True	DEPRECATED.These options are now maintained by the Documanage Workstation Options dialog.

Sample ini file entry:

```
[DesktopImport]
WatchDirectory=C:\Documents and Settings\All Users\Desktop\Import to Documanage
PollingRateInMilliSeconds=5000
```

Domains

The Domains section stores the last used domain for the Documanage Workstation application.

Name	Possible Values	Default	Description
LastUsedDomain	Valid Domain Name	No Default	The name of the last domain used to log into the Documanage Workstation. This appears in the user credential's dialog at startup.
Domain<#>	Valid Domain Name	No Default	DEPRECATED

Sample ini file entry:

```
[Domains]
LastUsedDomain=mdldomain
```

WFDaemon

The WFDaemon section defines the runtime characteristics for the workflow daemon (such as its debugging information). The Workflow Daemon keeps a log of its activity. It fetches the settings for its log from the poffice.ini file.

Name	Possible Values	Default	Description
Interval	Positive Integer	1	The number of seconds to sleep between polling loops.
LogDetailLevel	0 = low 1 = medium 2 = high/extended	0	Specifies the detail level for log.

Name	Possible Values	Default	Description
LogExtraDetail	0 = Do Not Add Detail 1 = Add Detail	0	Specifies whether to log extra information with each entry or not. This information includes the module name and calling function.
LogDisable	0 = Allow Logging 1 = Disable Logging	1	Specifies whether to allow logging or disable it.
LogFile	Valid file path	"DMWorkDaemon.log"	Defines the name of the Workflow Daemon logfile.
LogMinSize	Size in Kbytes	100	Specifies the minimum amount of data to retain in the current log file when archiving (which happens with LogMaxSize is reached).
LogMaxSize	Size in Kbytes	200	Specifies the maximum amount of data to be added to the current log file before archiving.

Sample ini file entry:

```
[WFDaemon]
Interval=1
LogFile=wfdaemon.log
LogMinSize=100
LogMaxSize=200
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
```

DS_<Datasource Name>

The DS_<Datasource Name> section defines the connection credentials for the specified datasource (found in the EZPDSInfo section).

Name	Possible Values	Default	Description
UID	Valid DSN Username	No Default	The connection string's user ID value.
PWD	Valid DSN Password	No Default	The connection string's password value (encrypted & sanitized).
CON	Encrypted string	No Default	Datasource connection string
Connections	Positive integer	1	The number of ODBC connections to open to this LOB datasource.

Sample ini file entry:

```
[DS_DMGSAMPLES]
UID=sa
PWD=01tWg
```

PowerOfficeAdmin

The PowerOfficeAdmin section defines the runtime characteristics of the Documanage Administrator.

Name	Possible Values	Default	Description
DisplayUser		NTLM	
DisplayUserSwitch	TRUE | FALSE	TRUE	
DebugMode	1 = Record Errors 0 = Do Not Record Errors	0	If DebugMode is set equal to 1 then the POAdmin will record errors to a flat ASCII file (poadmin.log) in the same directory asPOAdmin.EXE. If the setting is not equal to 1, or missing, the file is not generated. By default, the log file is not produced.
DisplayODBCErrors	1 = Display ODBC Errors 0 = Do Not Display ODBC Errors	0	If DisplayODBCErrors is set equal to 1 then the POAdmin will pop up a message box displaying the DBMS errors. If the setting is not equal to 1, or is missing, no messages are displayed. This setting can be changed via the System Profile dialog. The option is set through Display ODBC Errors located at the bottom of the form. By default, no messages are produced.

Sample ini file entry:

```
[PowerOfficeAdmin]
DebugMode=1
DisplayODBCErrors=0
```

FileRooms

The FileRooms section has been deprecated.

Workflow<#>

The Workflow<#> section has been deprecated.

FileRoom<#>

The FileRoom<#> section has been deprecated.

PowerOfficePublisher

The PowerOfficePublisher section defines the runtime characteristics of the Documanage publisher application. It has been deprecated.

Name	Possible Values	Default	Description
PrinterDriverName	Valid driver name	0	Specifies the number of cabinets stored in this fileroom section of the ini file
PrinterDeviceName	Valid device name	No Default	Specifies a cabinet to be added to the tree control
PrinterPortName	Valid port name		

PowerOfficeInput

The PowerOfficeInput section has been deprecated.

Diagnostic tests

Refer to “Diagnostic Tests” on page 143 for information on how to configure the Documanage Server to execute diagnostic tests.

DmgBlob storage plugin

The Dmgblob plugin (dmgblob.dll) uses database BLOBs to store documents. It can write a detailed log of its activity. Logging is disabled by default. For performance reasons, logging should only be enabled to troubleshoot storage problems.

Name	Possible Values	Default	Description
LogFile	Valid file path	“dmgblob.log"	Defines the name of the Workflow Daemon logfile.
LogMinSize	Size in Kbytes	100	Specifies the minimum amount of data to retain in the current log file when archiving (which happens with LogMaxSize is reached).
LogMaxSize	Size in Kbytes	500	Specifies the maximum amount of data to be added to the current log file before archiving.

Name	Possible Values	Default	Description
LogDetailLevel	0 = low 1 = medium 2 = high/extended	0	Specifies the detail level for log.
LogExtraDetail	0 = Do Not Add Detail 1 = Add Detail	0	Specifies whether to log extra information with each entry or not. This information includes the module name and calling function.
LogDisable	0 = Allow Logging 1 = Disable Logging	1	Specifies whether to allow logging or disable it.

Sample ini file entry:

```
[DMGBLOB]
LogFile=dmgblob.log
LogMinSize=100
LogMaxSize=500
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
```

In this sample, the plugin will log at the highest level, with full detail.

DmgNTFS storage plugin

The DmgNTFS plugin (dmgntfs.dll) uses the disk file system of the native operating system to store documents in a directory for each configured volume. It can write a detailed log of its activity. Logging is disabled by default. For performance reasons, logging should only be enabled to troubleshoot storage problems.

Name	Possible Values	Default	Description
LogFile	Valid file path	“dmgntfs.log"	Defines the name of the storage plugin logfile.
LogMinSize	Size in Kbytes	100	Specifies the minimum amount of data to retain in the current log file when archiving (which happens with LogMax-Size is reached).
LogMaxSize	Size in Kbytes	500	Specifies the maximum amount of data to be added to the current log file before archiving.
LogDetailLevel	0 = low 1 = medium 2 = high/extended	0	Specifies the detail level (verbosity) for log.
LogExtraDetail	0 = Do Not Add Detail 1 = Add Detail	0	Specifies whether to log extra informa-tion with each entry or not. This informa-tion includes the module name and calling function.
LogDisable	0 = Allow Logging 1 = Disable Logging	1	Specifies whether to allow logging or to disable it.

Sample ini file entry:

```
[DMGNTFS]
LogFile=dmgntfs.log
LogMinSize=100
LogMaxSize=500
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
```

In this sample, the plugin will log at the highest level, with full detail.

VLAM storage plugin

The VLAMCmdr section sets up the VLAM plugin to communicate with the CommCommander server. VLAM, an acronym for Virtual Library Access Method, is a proprietary Skywire for Oracle storage layer that uses the Virtual Storage Access Method (VSAM) to store documents and their annotations on a mainframe running the MVS (Multiple Virtual Storage) operating system. The storage plugin provides access through Documanage and CommCommander to documents stored on VLAM chains on a host with an MVS operating system.

The VLAM storage plugin can write a detailed log of its activity. Logging is disabled by default. For performance reasons, logging should only be enabled to troubleshoot storage problems.

Name	Possible Values	Default	Description
CCINI	isi.ini	N.A.	The name of the CommCommander ini file.
Interval	Positive Integer	1	The number of seconds to sleep between polling loops.
LogFile	Valid file path	“dmgvlam.log"	Defines the name of the VLAM logfile.
LogMinSize	Size in Kbytes	100	Specifies the minimum amount of data to retain in the current log file when archiving (which happens with LogMax-Size is reached).
LogMaxSize	Size in Kbytes	500	Specifies the maximum amount of data to be added to the current log file before archiving.
LogDetailLevel	0 = low 1 = medium 2 = high/extended	0	Specifies the detail level for log.

Name	Possible Values	Default	Description
LogExtraDetail	0 = Do Not Add Detail 1 = Add Detail	0	Specifies whether to log extra information with each entry or not. This information includes the module name and calling function.
LogDisable	0 = Allow Logging 1 = Disable Logging	1	Specifies whether to allow logging or disable it.

Sample ini file entry:

```
[DMGVLAM]
CCINI=isi.ini
Interval=1
LogFile=dmgvlam.log
LogMinSize=100
LogMaxSize=500
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
```

DmgSNAP storage plugin

The DmgSNAP storage plugin (dmgsnap.dll) uses a file server that provides WORM-like storage with customizable expiration dates. It can write a detailed log of its activity. Logging is disabled by default. For performance reasons, logging should only be enabled to troubleshoot storage problems.

Name	Possible Values	Default	Description
LogFile	Valid file path	“dmgsnap.log"	Defines the name of the storage plugin logfile.
LogMinSize	Size in Kbytes	100	Specifies the minimum amount of data to retain in the current log file when archiving (which happens with LogMax-Size is reached).
LogMaxSize	Size in Kbytes	500	Specifies the maximum amount of data to be added to the current log file before archiving.
LogDetailLevel	0 = low 1 = medium 2 = high/extended	0	Specifies the detail level (verbosity) for log.
LogExtraDetail	0 = Do Not Add Detail 1 = Add Detail	0	Specifies whether to log extra information with each entry or not. This information includes the module name and calling function.
LogDisable	0 = Allow Logging 1 = Disable Logging	1	Specifies whether to allow logging or to disable it.

Sample ini file entry:

```
[DMGSNAP]
LogFile=dmgsnap.log
LogMinSize=100
LogMaxSize=500
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
```

In this sample, the plugin will log at the highest level, with full detail.

DmgEMCC storage plugin

The DmgEMCC storage plugin (dmgemcc.dll) connects Documanage to a file server that provides WORM-like storage with customizable expiration dates. It can write a detailed log of its activity. Logging is disabled by default. For performance reasons, logging should only be enabled to troubleshoot storage problems.

Name	Possible Values	Default	Description
LogFile	Valid file path	“dmgemcc.log"	Defines the name of the storage plugin logfile.
LogMinSize	Size in Kbytes	100	Specifies the minimum amount of data to retain in the current log file when archiving (which happens with LogMaxSize is reached).
LogMaxSize	Size in Kbytes	500	Specifies the maximum amount of data to be added to the current log file before archiving.
LogDetailLevel	0 = low 1 = medium 2 = high/extended	0	Specifies the detail level (verbosity) for log.
LogExtraDetail	0 = Do Not Add Detail 1 = Add Detail	0	Specifies whether to log extra information with each entry or not. This information includes the module name and calling function.
LogDisable	0 = Allow Logging 1 = Disable Logging	1	Specifies whether to allow logging or to disable it.

Sample ini file entry:

```
[DMGEMCC]
LogFile=dmgemcc.log
LogMinSize=100
LogMaxSize=500
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
```

In this sample, the plugin will log at the highest level, with full detail.

Sample POffice.ini file

A typical poffice.ini file is shown here.

```
[PowerOfficeRouter]
NetworkAddress=127.0.0.1
Protocol=ncacn_ip_tcp
Endpoint=4000
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
LogFile=DMGRouter.log
LogMinSize=100
LogMaxSize=500
LogSqlLevel=2
DeleteFileAtStartUp=0
Domain=MDLDOMAIN

[PowerOfficeServer]
NetworkAddress=127.0.0.1
Protocol=ncacn_ip_tcp
Endpoint=4100
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
LogFile=poffice.log
LogMinSize=100
LogMaxSize=500
LogSqlLevel=2
DeleteFileAtStartUp=0
```

```
[EZPDSInfo]
DataSource=DMGSAMPLES
AuthID=dbo

[Directories]
Pictures=.\Bitmaps\

[StorageProviders]
NTFS=dmgntfs.dll
BLOB=dmgblob.dll
VLAM=dmgvlam21.dll:isi.ini
SNAP=dmgsnap.dll
EMCC=dmgemcc.dll

[AuthorizationProviders]
DMGDB=DMGDB_Authorization.dll

[AuthenticationProviders]
Default=DMGDB_AuthenticationServer.dll

[PowerOfficeDesigner]
EnableNewTasks=1
RecentWorkflowCount=1
Item1=Claim Properties

[PowerOfficeClient]
MaxFolders=100

[DesktopImport]
```

```
WatchDirectory=C:\Documents and Settings\All
Users\Desktop\Import to Documanage
PollingRateInMilliSeconds=5000
UsePoller=1

[Domains]
LastUsedDomain=mdldomain

[DS_DMGSAMPLES]
UID=sa
PWD=01tWg

[WFDaemon]
Interval=1
LogFile=wfdaemon.log
LogMinSize=100
LogMaxSize=500
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0

[PowerOfficeAdmin]
DebugMode=1
DisplayODBCErrors=0

[DMGNTFS]
LogFile=dmgntfs.log
LogMinSize=100
LogMaxSize=500
LogDetailLevel=2
```

```
LogExtraDetail=1
LogDisable=0

[DMGBLOB]
LogFile=dmgblob.log
LogMinSize=100
LogMaxSize=500
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0

[DMGVLAM]
Interval=1
LogFile=dmgvlam.log
LogMinSize=100
LogMaxSize=500
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0

[DMGSNAP]
LogFile=dmgsnap.log
LogMinSize=100
LogMaxSize=500
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0

[DMGEMCC]
LogFile=dmgemcc.log
LogMinSize=100
```

```
LogMaxSize=500
LogDetailLevel=2
LogExtraDetail=1
LogDisable=0
```

Appendix C: Using the Documanage Service Controller

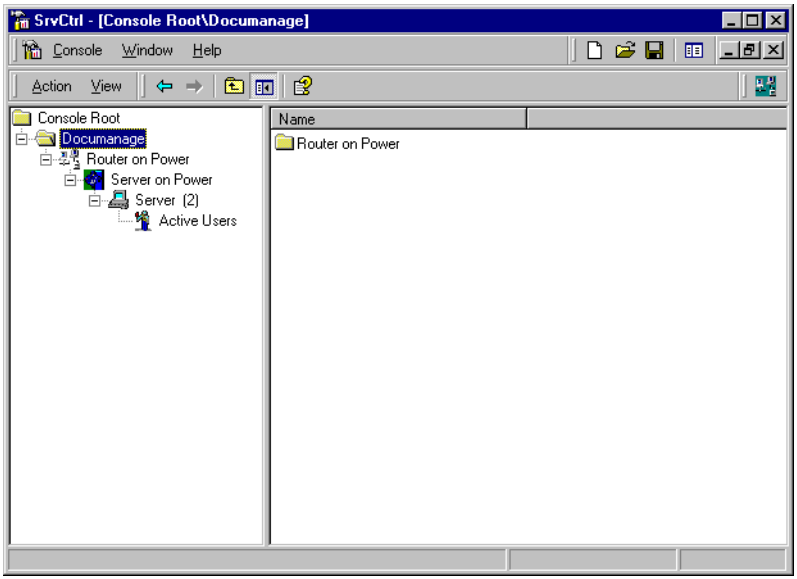
The Documanage Service Controller allows you to manage the Documanage Router and all Servers. This can be performed by any account with sufficient privileges, on any machine with network connections (to the Router and Server(s)). This is intended to facilitate remote administration.

NOTE: The Services controller can start and stop Server or Router services running only on Microsoft Windows. Unix-based Servers running on Unix or Linux operating systems cannot be started or stopped from the Service Controller.

You can use the Documanage Service Controller to inspect the operating state (and version level) of the Router and all Servers, and view and control usage. You can disconnect individual users, stop and start Servers, and stop and start the Router. These are potentially disruptive operations that should only be performed by knowledgeable administrators.

In the left pane of the Service Controller, the Router, Server, and Users are listed in a tree format. Selecting an icon in the left pane displays corresponding details in the right pane. For example, if the Server icon is selected on the left, a list of servers displays on the right.

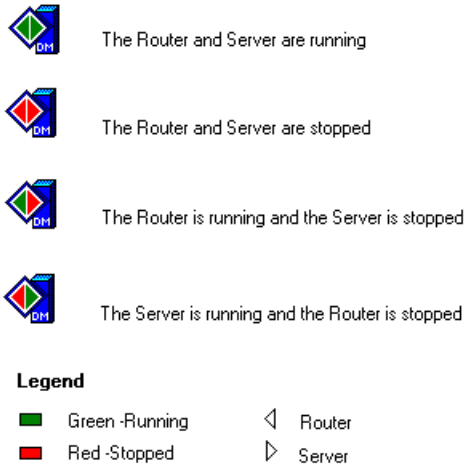
Right clicking on an icon in either pane yields an option to open a new window, displaying that branch of the tree and associated details.



Using the Documanage Service Agent

You can also monitor the operating state of the router and server on your computer by using the Documanage Service Agent. The service agent places a diamond shaped icon in the system tray on your desktop. This icon indicates if the router and server are running or not. The left side of the diamond represents the router and the right side represents the server. The colors green

and red indicate their operating status (running or stopped). You can right-click on the icon to change the operating status of the router and server.



The Documanage Router

The Router, which should be the first Documanage component started, acts as a Documanage “traffic cop,” coordinating the load of session connections between Clients and Servers by directing new Client connections to an available Server. This allows load sharing across multiple servers.

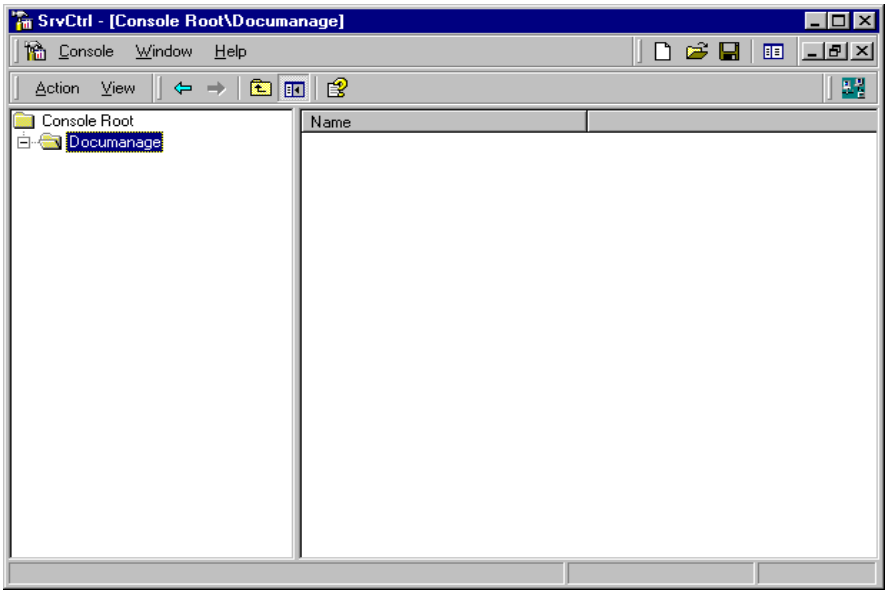
The Router is especially useful when there are multiple Documanage Servers, preventing bottlenecks and maximizing speed. For instance, if there are multiple Documanage Servers, the Router traffics information so that each Documanage Server handles an equal share of traffic.

The network settings used in the Connect to Router dialog and the Router Properties dialog must be the same as those set in the poffice.ini file on the Router machine. The .ini file tells the Router what protocols, address, and endpoint through which to listen. The radio buttons TCP/IP and Microsoft Networking on both dialogs correspond to the protocol settings (in poffice.ini) ncan_ip_tcp and ncacn_nb_tcb respectively.

You can start and stop the router, connect to server, update the service, and open a new window from the Service Controller.

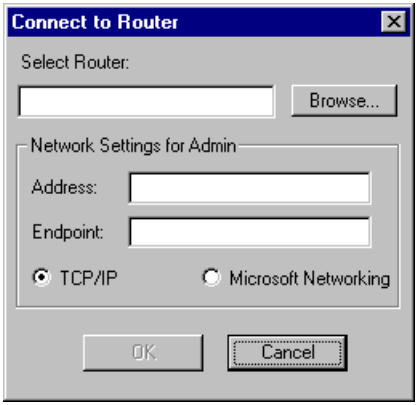
To add the Router

1 **Open the Documanage Service Controller (Start | Programs | Documanage | Documanage Service Controller.)**
You will see two folders on the left hand side of the dialog - Console Root and Documanage.



2 **Right-click the Documanage folder and select Connect.**

The Connect to Router dialog opens.



3 **Enter or browse to the Router and enter the Address and Endpoint.**

- **Address:** This is the name or TCP/IP address of the Router.
- **Endpoint:** This is TCP/IP network specific information. Enter 4000 unless otherwise stated. Check with your network administrator if you have any questions.

4 **Click OK.**

The Router has been added. You can now start, stop, and check router properties from here.

To start the Router

You can start the Documanage Router by highlighting the Router in the Service Controller, right-clicking and selecting *Start*, or by highlighting the Router and clicking the traffic light icon in the top right of the dialog.

The Documanage Server

The Documanage Server stores and retrieves information in Documanage. Documanage supports multiple servers, which run on different machines. They are displayed as Server on... in the Service Controller. Multiple instances of a server are run on a single machine and are listed beneath each Server. Server instances reflect refresh operations from the Administrator.

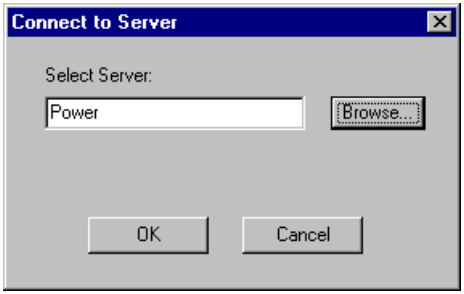
The Server instance list is refreshed automatically every few minutes, but can be refreshed manually.

To add the Server

1. **Open the Documanage Service Controller. (Start | Programs | Documanage | Documanage Service Controller.)**

2. **Right-click the Router and select Connect to Server.**

   The Connect to Server dialog opens.



3. **Enter or browse to the Sever and Click OK.**

   The server has been added. You can now start, stop and check server properties from here.

To start the Server

You can start the Documanage Server by highlighting the Server in the Service Controller, right-clicking and selecting *Start*, or by highlighting the Server and clicking the traffic light icon in the top right of the dialog.

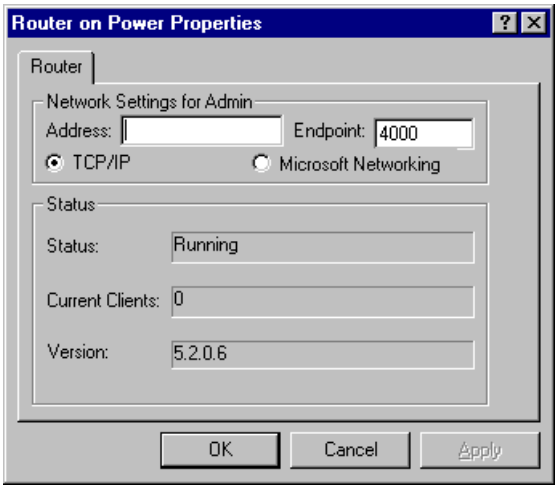
To check Router and Server properties

1. **Open the Documanage Service Controller. (Start | Programs | Documanage | Documanage Service Controller.)**

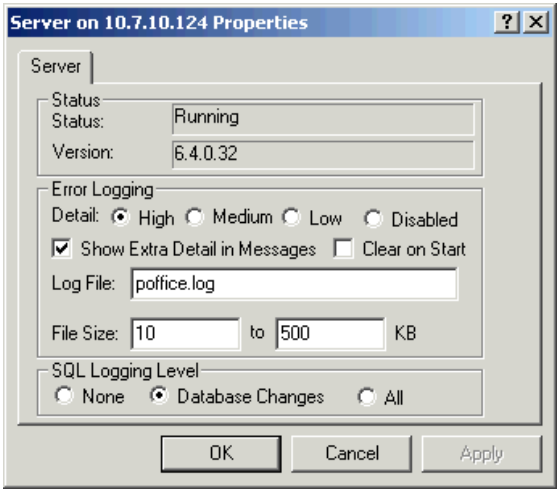
2. **Right-click either the Server or the Router depending on the properties you want to see and select Properties.**

   The Properties dialog open.

   - **Router Properties:** In the Router Properties dialog you can see the Network Setting and Status information.



- **Server Properties:** In the Server Properties dialog you can see Status and Error Logging information.



Restarting the Documanage Server

The Documanage Server needs to be restarted whenever a change that is made and saved in the Documanage Administrator needs to display in the Client.

Clicking the *Exit* button from the main Administrator dialog or clicking *Refresh* from any dialog menu will prompt the user to refresh the server.

Server status is reflected in the Server *Manager Report* dialog, which appears when the Server is refreshed. When the Client machine is disconnected, all previous connections to old Servers (i.e., those marked as “stale”) are terminated and those Servers are shut down. Additionally, once a Server is

refreshed, no new connections can be made to an old Server (i.e., those marked as “stale”). The dialog then closes automatically.

NOTE: If you access an Administrator function and exit the Administrator without making any changes, you will not be prompted to refresh the Server as no changes were made.

Disconnecting a user from the Documanage Server

To disconnect a user from a Documanage Server, select *Disconnect User* from the *Options* menu in the Documanage Server. For example, if a user on a remote site forgets to logout of Documanage, their session can be closed from the Server.

To disconnect a Documanage user from the server

1. **Go to the Service Controller and make sure the Server is running.**
2. **Expand the tree so the list of users is displayed.**
3. **Highlight the user you want to disconnect and click *Disconnect.***

Appendix D: VLAM Storage

Introduction

VLAM, an acronym for Virtual Library Access Method, is a proprietary Skywire for Oracle storage layer that uses the Virtual Storage Access Method (VSAM) to store documents and their annotations on a mainframe running the MVS (Multiple Virtual Storage) operating system. It provides access through Documanage to documents stored on VLAM chains on MVS. This appendix describes VLAM storage for Documanage.

Libraries

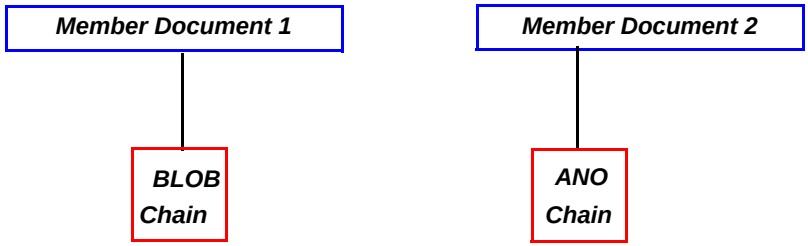
Virtual Library Access Method uses named *libraries* containing collections of named *members*. Documanage uses a member to represent a single instance of a document or the annotations for a document. A *Chain* in the member stores for a document and the annotations for a document. The documents themselves are stored in a member using a chain named BLOB and their annotations are stored in a separate member using a chain named ANO or ART. To represent a set of documents or annotations, three things are thus required:

- a library name
- a member name
- a chain name

An Example

In the example shown here, the library name is Sample, the member names are Member Document 1 and Member Document 3. The chain names are BLOB chain, and ANO chain.

Sample

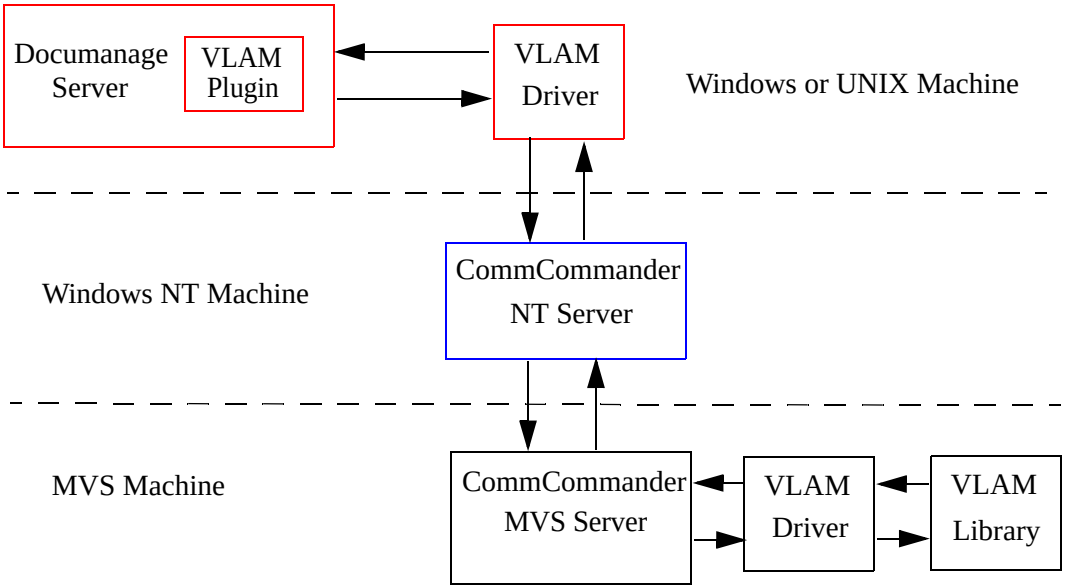


This library contains two members; each has a one chain.
The library identifies the chains by name.

VLAM Architecture

VLAM interfaces the Documanage Server to MVS systems so that it can store documents in VLAM libraries. The Documanage Server accesses VLAM via Skywire for Oracle's CommCommander application. CommCommander consists of two modules: one runs on MVS and the other runs on Windows NT. A simplified block diagram of the VLAM architecture is shown here.

The Documanage Server connects to a VLAM driver on its machine via its VLAM plugin . This VLAM driver connects with the CommCommander NT server which in turn connects to the CommCommander MVS server. The CommCommander MVS server uses its VLAM driver to retrieve data from the VLAM library. It sends the data back to the CommCommander NT Server, which in turn sends it back to the Documanage Server via its VLAM driver and plugin.


Documanage Server
VLAM Plugin
VLAM Driver
Windows or UNIX Machine
Windows NT Machine
CommCommander NT Server
MVS Machine
CommCommander MVS Server
VLAM Driver
VLAM Library

Naming Conventions

VLAM uses naming conventions for its Libraries, member names and file extensions.

Libraries

VLAM libraries are specified as “VLAM|<libraryname>” in the database table OT_VOLUMES. For example, the library VLAMVOL1 would be specified as “VLAM|VLAMVOL1” where “VLAM|” is the storage type and “VLAMVOL1” is the library name.

Member Names

Documanage assigns member names when it adds new documents to the system. The full file path is formed by adding the member preceded by a "\." The member name is a 10-digit number left-filled with zeros. For example, for member 0000001234.doc, the full file path is "VLAM|VLAMVOL1\0000001234.doc." The annotation data would be stored in a separate member whose full path is VLAM|VLAMVOL1\000001234.ano.

File Extensions

The Documanage server uses the .doc extension while VLAM member chains are BLOB and ANO chain names. The BLOB chain holds the document data and the ANO chain (if present) holds any annotations for the document. The VLAM plugin handles the naming conversion when it imports files into Documanage or exports files back into VLAM. The document data goes into a member called EZPDOCID:0000001234, in the BLOB chain. Annotation data for the document goes into a member called ANO000001234 in the ANO chain.

Storage Volumes

Use the categories screen in the Documanage Administrator to set up storage volumes:

- Since Documanage uses VLAM to archive documents and their annotations, we recommended that you send files to VLAM for permanent storage and send files to local magnetic media (a hard disk drive for example) for temporary storage. This stores a viewed or checked out document on local media and then stores it on VLAM when it is saved and closed.

- Because annotation files change frequently, it is preferable to store them on a magnetic storage volume. Use the System dialog in the Documanage Administrator to assign Annotation files to magnetic storage.

Setting Up Categories

You can use Categories in the Documanage Administrator to reference VLAM libraries as document storage locations.

In the Documanage Administrator program the storage location will be defined as **VLAM|ddname**. **VLAM|** indicates that the storage location is a VLAM library and the **ddname** contains the name that Documanage uses to reference the VLAM library.

Documanage uses a table of DD names and the VLAM libraries they reference. The DD name defined in the Documanage Administrator must agree with the DD name supplied to Documanage.

For example, suppose we use the Documanage Administrator to set up a Document Category called **Claims**. The storage for this Document Category is defined to be **VLAM|VLAMVOL1**. When a document has a Document Category of **Claims**, Documanage stores the document in the VLAM library referenced by the DD name **VLAMVOL1**.

When you view this document, the Workstation requests the document from Documanage because of the **VLAM|** prefix. Documanage then retrieves the document from the VLAM library associated with the DD name **VLAMVOL1** in its setup table.

Appendix E: EMC Centera Settings for Documanage

The Documanage EMC Centera API Integration product described in this appendix is available under a separate license from Oracle Software. It is not a part of the Documanage product.

References in this documentation to the Documanage EMC Centera API Integration product, its configuration, or its interfaces to the Documanage product do not imply that the Documanage EMC Centera API Integration product has been incorporated into the Documanage product.

Since your software license may not include the Documanage EMC Centera API Integration product, the software discussed in this appendix may not be a part of your installation.

Contact your Oracle Software sales representative if you wish to license the Documanage EMC Centera API Integration product.

EMC Centera must be set up in a certain way to operate properly with Documanage. Also, Documanage does not support all Centera options and capabilities related to document retention. This appendix only describes settings related to the operation of Documanage features and it only recommends settings that have an influence on the behavior of Documanage. The Centera settings that are related to Documanage include cluster settings and replication/fail over settings.

NOTE: This appendix does not describe how of EMC Centera operates. Refer to the EMC Centera documentation learn about the concepts, capabilities and features of EMC Centera that you need to make these settings. You should rely on EMC field-engineers and sales engineers for the best configuration of your Centera system. Documanage places minimal requirements on the configuration and connectivity of

your Centera system; this allows it to have the best possible performance with the widest possible array of configurations.

Centera Settings

Centera cluster settings and replication/failover settings must be made to accommodate Documanage.

Cluster Settings

Cluster settings include application profile settings, the compliance mode, retention classes, the retention period, and constraints.

Application Profiles

Documanage requires *read*, *write*, *delete* and *exist* access rights to the Centera. Documanage uses a configurable string to access each Documanage volume defined on Centera. Typically, you use one volume name to refer to an entire Centera. This string may be configured with multiple primary and secondary cluster addresses as well as a PEA (Pool Entry Authorization) file.

Compliance mode

The compliance mode must be CE (GE) or CE+. Documanage retention management capabilities are available in every installation. To use them with Centera storage and to have retention enforced by the Centera system, the Centera must use one of the compliance-supported models.

Retention Classes

Documanage does not use Centera retention classes and it never sets them in Documanage documents. This setting should not impact Documanage operation.

Retention Period (default)

Documanage sets an explicit retention period on every document, even if the period is zero. As such, this setting should not affect Documanage.

Constraints

- **check_retention_present**—Documanage sets an explicit retention period on every document, even if it is zero. Since this is the case, check_retention_present should not affect Documanage.

- **check_retention_range**—check_retention_range specifies the minimum and maximum retention periods. If check_retention_range is set, all Documanage document categories must be defined with retention periods that fall within this Centera-configured range.

  If no retention is configured in Documanage for a document Category, such documents will be set with a zero retention interval. If this or any other explicit retention setting falls outside the Centera constraint setting, an error will occur. Documanage cannot override Centera compliance/retention constraints.

NOTE: How this setting is applied may be changing in the Centera interface. Later Help documentation shows this under the command **update pool retention <name>**. The principles described above still apply.

Replication/Fail-over Settings

EMC provides you with a considerable amount of documentation and consulting on how to configure Centera for different replication and failover strategies. EMC will help you come up with a configuration that meets your needs. Oracle is unable to make any recommendations on how to configure Centera for replication and fail-over.

The EMC SDK controls the behavior of Documanage using the contents of the connection address string and the associated PEA file. The connection address string for each volume in the Administrator program is defined to Documanage as the Volume Path.

Appendix F: Network Appliance Settings for Documanage

If the value of the snaplock_maximum_period for a SnapLock-enabled volume is less than infinite, and the volume is associated with a Documanage Category configured for retention forever or for retention beyond the snaplock_maximum_period, then files associated with the Category will be be retained on the volume only for the snaplock_maximum_period.

When the snaplock_default_period exceeds the snaplock_maximum_period, the SnapLock-enabled volume applies the lesser value, which is the maximum retention period. For instance, if the snaplock_maximum_period is 30y, and the snaplock_default_period is infinite, then files associated with the Category will be stored in the volume with protection enforced for no more than 30 years.

Documanage cannot detect these hardware protection shortfalls. You must ensure that the snaplock_maximum_period and snaplock_default_period values are configured to avoid them.

To allow Documanage to provide storage protection with a duration of forever on a SnapLock-enabled volume in a Network Appliance System, you must set the volume configuration of the system to make the snaplock_maximum_period value infinite. You can use the Data ONTAP administration console to set the snaplock_default_period to infinite, without changing the snaplock_maximum_period or having the console produce error messages.

Setting the snaplock_default_period to infinite will not enable infinite duration WORM status, because the maximum retention period takes precedence over the default retention period, and the factory-shipped value for snaplock_maximum_period is 30y (30 years).

Use the Data ONTAP administration console command shown here to set the default configuration of a SnapLock enabled volume. This command makes infinite-duration WORM status available for files initially stored using a Documanage Category configured for protection forever.

```
filer> vol options SnapVol snaplock_maximum_period
infinite
```

NOTE: In these examples, the console prompt is `filer>`, and the volume name is `SnapVol`. For your configuration, please use the same commands and substitute your volume name in place of `SnapVol`.

Appendix G: LDAP Authentication and Authorization

Documanage can be configured to utilize an existing Lightweight Directory Access Protocol (LDAP) service to supply authentication and group membership definitions for authorization. This capability is provided by two LDAP plug-in modules included in both the Documanage Server and Router folders:

dll	function
Ldap_authenticationserver.dll	Server-side authentication services
Ldap_authorization.dll	Server-side group resolution services for authorization

Documanage integration with your LDAP infastructure is flexible, but proper configuration requires understanding of what Documanage needs from LDAP and how your LDAP schema is shared with Documanage. A section "[LDAPAuthConfig]" in the POFFICE.INI file configures the plug-ins with the required schema element names. Two basic LDAP objects are required: one object represents users and provides the means to authenticate them and a second represents groups which can be used to assign Documanage

authorities. How these objects are found in the LDAP system and the names of the various required attributes is completely configurable.

NOTE: All following POFFICE.INI entries, unless otherwise noted, are found in the [LDAPAuthConfig] section of the POFFICE.INI. Please see Appendix B for more information on the POFFICE.INI file.

Binding to the LDAP Service

Documanage binds to your LDAP service according to the configured parameters, and a server name or address and port is required. A Distinguished Name (DN) and password for the bind must be configured if your LDAP service has security enabled and requires an authenticated bind to the service for querying User and Group objects. Documanage treats LDAP as a read-only data source and never writes or updates the LDAP records. In summary, the following settings in the Poffice.INI provide the binding information:

Name	Value
Host	Hostname or IP address of LDAP server
Port	Port number of LDAP server on host
Binddn	DN to use for authenticated connection
Bindpw	Password for binddn account

Search Base and Scope

The search base determines where in your LDAP database tree Documanage starts searches for User and Group objects. The search scope parameter controls the depth of searching within the tree. The base object is identified by a DN which is effectively added to all search criteria. The scope can be one of three values: “base”, “one” or “sub”, described in the table below:

Name	Description
Base	The distinguished name (DN) of the entry at which to start the search
Scope	**base**: Search the base-entry only. **one**: Search all entries in the first level below the base-entry, excluding the base-entry **sub**: Search the base-entry and all entries in the tree below the base

Users

Documanage requires three attributes for User objects: a user ID, a user’s “friendly name” and a password. Documanage authentication uses a challenge-and-response protocol which requires an MD4 hash of the user’s password to be available at the server. For this reason, User objects used for Documanage authentication must have either an MD4 hash or clear-text password attribute, or both. An MD4 hash *must* be stored as a 32-character

string of hexadecimal digits which represent the value of the 16-byte binary MD4 hash.

Skywire's Documanage for Oracle has supplied a suitable auxiliary objectclass that adds an MD4 password attribute to a User object. Within the 'docucorp.schema' file supplied with Documanage, the 'docucorpAccount' objectclass adds the 'doccNTPassword' attribute with a suitable definition to hold an MD4 hash converted to a hexadecimal string. Further, Skywire's Documanage for Oracle supplies a password editing plug-in for the JExplorer open-source LDAP browser. This plug-in recognizes the 'doccNTPassword' attribute name and provides the user the ability to type in a clear-text password and have it properly MD4 hashed and converted to a hex string. Note that since MD4 is a one-way hash algorithm, passwords *cannot* be recovered from the database once stored this way.

To improve efficiency, some LDAP schemas also provide for User objects to list the groups of which they are members. This multi-valued attribute is generically called a "member of" attribute because it provides a list of all the groups a user is a "member of." Groups are referenced in "member of" values by their LDAP Distinguished Name, or "DN."

NOTE: If your configuration indicates that your LDAP system supports the use of a "member of" attribute, the attribute must be created and maintained by your LDAP administration software. Documanage will expect the information to be correct. For example, Documanage will report that a user is not a member of any groups if there are no "member of" attributes, regardless of actual group object contents. To

force Documanage to refer exclusively to group objects to determine group membership, do not configure the “member of” attribute name.

Here is a summary of all User-object-related configuration parameters and attributes from the POFFICE.INI file:

Name	Values
uidfilter	An arbitrarily complex filter which limits searches to only desired User objects. This expression should be surrounded with parentheses to fit into an overall expression of the form: (&((<uidattr>=userid)<uidfilter>) This expression may be empty.
uidattr	The name of the User object attribute that contains the User ID
uidfriendlyattr	The name of the User object attribute that contains the User’s full name
uidpw	The name of the User object attribute that contains the password followed by a comma and the encoding scheme for that password attribute. More than one pair may be specified by separating them with a vertical-bar. In this case, the attributes are processed in the order listed as the order of preference for which password to use. Two formats are supported: **MD4** and **clear**.
uidgroupattr	The name of the User object attribute (multi-valued) that contains the DN of each group of which this user is a direct member. This attribute should NOT be configured if your LDAP system doesn’t maintain these “member of” references.
uidgroupmemberclass	The Organizational Unit of the Group objects referenced by DN’s in the attribute named by ‘uidgroupattr’.

Groups

Documanage uses Group membership to confer authorities. Authorities are assigned to group names in the Administrator module and users inherit those authorities through membership in the group. Groups may have both Users and other Groups as members. When a user is a member of a group and that group is in turn a member of another group, the user is considered a member of both groups. This may continue for any number of levels.

If the “uidgroupattr” User attribute is configured in the Poffice.ini file, first-level immediate group memberships for a given user are determined from this User attribute. These groups are then resolved for any groups of which they are members, and so on. If the “uidgroupattr” attribute is *not* configured or blank, then all groups are searched for the user’s DN to get the first-level immediate group memberships. Parent-group resolution then continues normally.

Here is a summary of all Group-object-related configuration parameters and attributes from the POFFICE.INI file:

Name	Values
groupfilter	An arbitrarily complex filter which limits searches to only desired Group objects. This expression should be surrounded with parentheses to fit into an overall expression of the form: (&((<groupnameattr >=groupname)<groupfilter>) This expression may be empty.

Name	Values
groupnameattr	The name of the Group object attribute that contains the Group ID
groupfriendlynameattr	The name of the Group object attribute that contains the Group's user-friendly description
groupmemberattr	The name of the Group object attribute (multi-valued) that contains the DN of each User or Group object which is a direct member of this group
groupuidmemberclass	The Organizational Unit of the User objects referenced by DNs in the attribute named by 'groupmemberattr'. Other members are expected to be Groups.

An Example

Example Company (example.com) has a simple LDAP structure. Example.com has objects for Users and Groups with these objectclasses, separated into their own organizational units:

- **User objects** are defined as objectclasses '*person*' and '*docucorpAccount*':
  - '*person*' provides attributes of 'cn' and 'userPassword'
    - '*cn*' holds the user's common, or "friendly", name
    - '*userPassword*' may hold the user's clear-text password
  - '*docucorpAccount*' provides the attributes 'userid' (or 'uid') and 'doccNTPassword'
    - '*userid*' (or 'uid') is the log-on name of the user

- '*doccNTPassword*' holds an MD4 password hash

- **Group objects** are defined as objectclass '*groupOfUniqueNames*'
  - '*groupOfUniqueNames*' provides attributes of '*uniqueMember*' and '*cn*'
    - '*uniqueMember*' holds the distinguished name (DN) of the user or group that is a member, one for each member
    - '*cn*' holds the common, or descriptive user-friendly, name of the group

So, the structure of Example.com company, with separate OU groupings for users and groups, may be defined as follows:

```
## ---> Root Base Object for Example.com
##
dn: dc=example,dc=com
objectClass: dcObject
objectClass: organizationalUnit
dc: example
ou: Example Company Base Object

## ---> Structural Objects - The 'users' ou
##
dn: ou=users,dc=example,dc=com
objectClass: organizationalUnit
ou: users

## - The 'groups' ou
```

```
##
dn: ou=groups,dc=example,dc=com
objectClass: organizationalUnit
ou: groups
```

---> The LDAP user with which we can bind to the LDAP server

```
##
## User: ldapuser
dn: uid=ldapuser,dc=example,dc=com
objectClass: person
objectClass: uidObject
objectClass: simpleSecurityObject
cn: LDAP User
sn: Administrator
description: Sample Docucorp admin account
uid: ldapuser
userPassword: bindpassword
```

Here are a couple simple examples of defined objects:

---> Sample Users - User: dmguser

```
##
dn: uid=dmguser,ou=users,dc=example,dc=com
objectClass: person
objectClass: docucorpAccount
uid: dmguser
userPassword: secretword
doccNTPassword: 5D77280F6936E92F41686198C870A4CF
description: Sample Docucorp user account
cn: Test User
```

```
sn: TestUser


## - User: demo
##
dn: uid=demo,ou=users,dc=example,dc=com
objectClass: person
objectClass: docucorpAccount
uid: demo
userPassword: secretword
doccNTPassword: 5D77280F6936E92F41686198C870A4CF
description: Documanage Demo Account
cn: Demo Account
sn: DemoAccount


## ---> Sample Group - dmgusers
##
dn: cn=dmgusers,ou=groups,dc=example,dc=com
objectClass: groupofUniqueNames
ou: groups
cn: dmgusers
description: Documanage Users
owner: uid=dmguser,ou=users,dc=example,dc=com
uniqueMember: uid=dmguser,ou=users,dc=example,dc=com
uniqueMember: uid=demo,ou=users,dc=example,dc=com
```

Poffice.ini Settings for Example.com

To now utilize the above LDAP schema and objects for Example Company in Documanage, the following section in the POFFICE.INI must be added and configured. For Example.com, it would look like this:

```
;LDAP Settings
[LDAPAuthConfig]
Host=LDAPServerHostName
Port=389
Binddn=uid=ldapuser,dc=example,dc=com
Bindpw=bindpassword
Base=dc=example,dc=com
Scope=sub
Uidattr=uid
Uidpw=doccNTPassword,MD4|userPassword,clear
Uidfriendlyattr=cn
Uidfilter=(objectClass=docucorpAccount)
Uidgroupmemberclass=groups
Groupfilter=(objectclass=groupOfUniqueNames)
Groupnameattr=cn
Groupmemberattr=uniqueMember
Groupfriendlynameattr=description
Groupuidmemberclass=users
```

Appendix G: LDAP Authentication and Authorization

Index

E

I

L

M

O

P

R

S

T

U

V

W