

Oracle® Documanage

**General Reference for the
Documanage Bridge**

version 3.5

Part number: E14904-01

April 2009

Copyright © 2009, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Oracle, JD Edwards, and PeopleSoft are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

PUBLICATION HISTORY

First issue for Version 3.0: February 2005

Revision 1 for Version 3.1: September 2005

Revision 3 for Version 3.1 Service Release 1: January 2006

Revision 4 for Version 3.2 Service Release 1: August 2006

Revision 5 for Version 3.3: September 2006

Revision 6 for Version 3.4: September 2007

Revision 7 for Version 3.4: May 2008

Revision 8 for Version 3.4: March 2009

Contents

iii Publication history

Chapter 1, Overview

- 2 Skill Requirements
- 4 System Architecture
- 5 Platforms
- 6 System Requirements

Chapter 2, Installing the Bridge

- 8 Introduction
- 9 Components
- 10 Requirements
 - 10 Hardware
 - 10 Software
- 11 Installation
 - 11 Graphical Installation on Windows and Unix
 - 18 Terminal Installation on UNIX Operating Systems
 - 19 Using the Installation Script
- 23 Uninstalling the Bridge
 - 23 Uninstallation from Windows
 - 23 Uninstallation from UNIX

Chapter 3, Creating a Bridge Application

- 27 Server Rules Lists
- 28 Creating Web Pages
- 29 URL Encoding
- 30 Bridge Encoding

31	UI Configs
32	User Security
32	Login
33	BridgeURLs and Secure Login
33	Automatic Logout when Quitting the Browser
34	The Bridge Cookie
35	ReqTypes
36	Redirection
37	Ad Hoc Logins
38	How Server Rules Signal an Error
39	Dates as Inputs and Outputs
40	NULL Values as Inputs and Outputs
41	SQL Builder Syntax
41	Filter Syntax
45	Filter Operators
47	Orderby Syntax
49	Downloading Files from the Client Application
50	Serving Files from the Web Server
51	Transferring Files from the Bridge Server to the Client Server
51	Choosing a File Transfer Method
51	Configuring Shared Directories
51	Configuring the TempCache Directory
51	Configuring the FileCache Directory
53	Transferring Files Via the Queue
55	Converting Documents
56	Rendering Page Images
57	Property Sets and Property Set Lists
57	Property Sets
59	Property Set Lists
60	Compound Documents
61	Non-Web Applications

Chapter 4, Configuring the Documange Bridge

64 The Dmg_Brs INI File

- 64 [Logging]
- 65 [INIsToLog]
- 65 Configuring the Bridge Account
- 66 [BridgeProfile]
- 66 [Documange]
- 67 [BridgeCredentials]
- 67 [UserCredentials]
- 69 [BridgeURL]
- 70 [ValidateSession]
- 71 [FileCache]
- 72 [TempCache]
- 72 [CdPlugins]
- 72 [ReqTypes]
- 73 [PropertySets]
- 73 [NullValues]
- 74 [Web]
- 74 [Query]
- 75 [QueryDocuments]
- 75 [QueryProjects]
- 76 [QueryAllProjects]
- 76 [QueryDocumentRenditions]
- 77 [ListUsers]
- 77 [PresentDocument]
- 77 [PresentCompoundDocument]
- 77 [PresentSubdocument]
- 78 [ReturnDocument]
- 78 [DocsIndicator]
- 78 [CreateDocument]
- 78 [UpdateAnnotations]
- 78 [CheckOutDocument]
- 79 [CheckInDocument]
- 79 [PresentFolder]
- 79 [SetupFolder]
- 79 [DeleteFolder]
- 79 [PresentProject]
- 80 [GetTaskQueue]
- 80 [Notes]

	80	[CreateDiary]
	81	[QueryDiaries]
	81	[Rendering]
	82	[ReturnDocumentPage]
	82	[ReturnConvertedDocument]
	83	[ThumbNails]
	84	[DmgDiagnostics]
85		The Dmg_Brmime INI File
86		The Dmg_Brc INI File
	86	[Logging]

Chapter 5, Troubleshooting and Diagnostics

88	Bridge Logging
88	IDS Logging
88	Assessing Bridge Component Versions
88	Documanager Diagnostic Tests
88	Distinguishing Bridge Problems From Documanager Problems
89	Distinguishing Bridge Server from Bridge Client Problems

Chapter 1

Overview

Through Docupresentment (IDS), Documanager Bridge provides access to the power of Documanager.

With this bridge, you can web-enable Documanager to provide such capabilities as:

- Searching for documents, folders, projects, notes, or diaries
- Retrieving document contents files or pages of Document contents files
- Retrieving and updating document annotations files
- Presenting the properties of documents, folders, projects, notes, or diaries
- Adding new documents, folders, projects, notes, or diaries
- Updating documents, folders, projects, notes, or diaries
- Updating document contents files
- Deleting documents, folders, projects, or diaries
- Using workflow, including managing tasks, advancing projects, and more

The bridge is like an API or a tool kit. It provides the tools that you need, in the form of IDS rules, to build an application based on IDS that can do virtually anything that can be done with Documanager.

The primary component of Documanager Bridge is a library that contains all of the bridge rules. In addition, there are some configuration files that these rules use. Describing the library, the rules, and the configuration files is the main topic of the bridge documentation set.

SKILL REQUIREMENTS

[Creating a Bridge Application on page 25](#) provides an introduction to writing an application with the bridge. It assumes that you already know about the environment in which the bridge exists. This documentation is oriented toward the bridge application developer, and the administrator of a bridge system. It is the responsibility of the bridge application developer to define the skill requirements for any user of the bridge applications that they may develop.

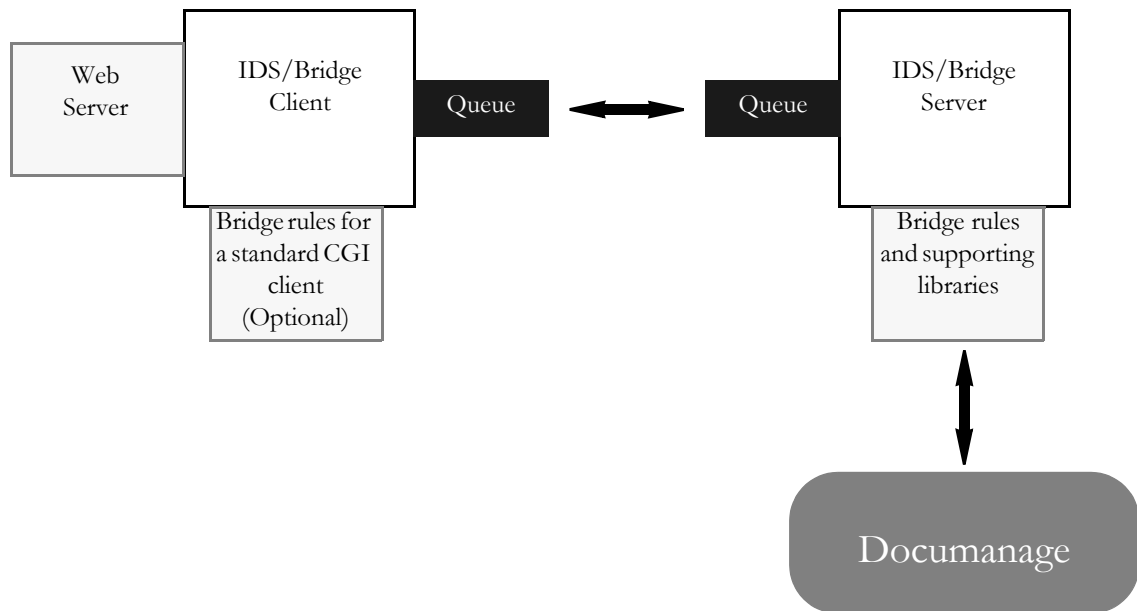
- Basic Skills—Perform basic-level platform skills, such as:
 - Start applications
 - Establish a network connection
 - Navigate the desktop using the mouse and keyboard
 - Find, open, print, copy, and move files in directories / folders / volumes
 - Understand where to find basic program features
 - Understand and use product documentation
 - Determine the version information of installed software applications
 - Determine hardware configuration and free space on volumes
- Role: Administrator—Must have basic skills, and additionally be able to perform intermediate-level and advanced-level platform skills, including:
 - Install, remove, and set up applications.
 - Install, remove, set up, and administer IDS, and queues and other supporting software used with IDS.
 - Install, remove, set up and administer any required web servers.
 - Install, remove, set up and administer Documange, including the use of the Documange Administrator application.
 - Use the Documange Workstation.
 - Identify general hardware, application, system, and network errors
 - Access, preserve and transmit log files as requested by Skywire Software Support
 - Determine and report what software version of the operating system is installed, including what patches or service packs have been applied.
 - Determine and report what application software packages and versions have been installed and which are executing (including unattended processes).
 - Have rights to install, configure and remove software in system directories and to administer user accounts
 - Install and set up operating systems for platform as noted below under Software Requirements
 - Perform significant software and network troubleshooting
 - Administer and configure networks (TCP/IP, routers, remote access)
 - Perform system backups

- Install, start, stop, and configure auto-start unattended processes (for example, UNIX daemons, Windows (NT) Services) as required.
- Locate and interpret errors in application logs and system event logs
- Locate and modify INI files
- Create and edit registry keys on Windows
- Configure users, local groups, and global groups for an individual machine and across domains
- Modify permissions/authorities for files and directories
- Stop, start, and configure database management servers
- General Hardware Skills—Must also be able to perform general hardware skills, such as:
 - Optimize hardware configuration (install / format hard drive, install /configure RAM, I/O cards, PCI, ISA, NUBUS)
 - Set up and troubleshoot scanners, printers, and optical devices
 - Install and/or replace major components (hard drive, CD-ROM, video board, RAM, power supply)
 - Perform minor hardware repairs and preventive maintenance
 - Configure SCSI connections/termination
- Role: Developer—Must have basic skills. Must have all skills of the Administrator role or have access within their organization to such a person, and additionally have software development skills, including the ability to:
 - Install and set up target platform development tools
 - Understand the header files for the Documange API.
 - Develop an IDS client application, including formulating rules lists, using IDS standard, server, and client rules, calling the DSI interface, writing an IDS rule if required, and communicating with other applications such as web servers.
 - Understand the domain and requirements of the bridge application
 - Use the standard IDS client, the DSICo Test Bed, or other applications used to provide standard testing of IDS transactions.
 - Program in the bridge application target development environment, such as ASP, JSP, or C.
 - Use HTML and HTTP sufficiently well to build an IDS client application, if the application is to be web-enabled.
 - Develop application installers appropriate to target platform

SYSTEM ARCHITECTURE

With Documanager Bridge, the IDS client submits requests to the IDS server, which uses bridge rules to process them. The bridge server rules handle all communication with Documanager. The IDS server submits the results of requests to the IDS client, which then forms a response for the user.

In the following diagram, the IDS client and server are labeled *IDS/Bridge* Client and Server because they are customized to handle transactions that access Documanager. Bridge components consist primarily of rules that run on the IDS server, plus any required supporting libraries. If you use the IDS standard CGI client application, there are some bridge rules that you will find of use there as well.



The IDS client is typically a web server application, although that is not required. It may be a standalone application, or a component of some other type of system. In most cases, the IDS client and server are on separate server machines, connected by a network and running queue software to communicate with each other. Documanager will also typically be on its own server machine or machines, and must be connected to the IDS server machine by a network.

PLATFORMS

The library of Documanage Bridge rules is available for:

- Windows 2000 Server or higher
- Solaris 7 or higher
- AIX 4.3 or higher
- Redhat Linux AS 2.1 or higher

This release of Documanage Bridge rules requires Documanage version 6.4 or newer.

A few capabilities provided by the bridge server rules are available on Windows only. These include:

- Converting Documents to image PDF (with DmgBrsReturnConvertedDocument)
- Converting FPP to XML (with DmgBrsReturnConvertedDocument)
- Creating images of individual pages of Documents (DmgBrsReturnDocumentPage, DmgBrsReturnThumbnail)
- Decompressing DCZ files (DmgBrsPresentDocument, DmgBrsPresentDocumentPage, any Bridge server rule whose name begins with "DmgBrsReturn")

Different components of the system that you can run on separate server machines can be on different platforms. For example, Documanage and IDS could be on Windows servers, while the IDS client and web server could be on a Linux server.

SYSTEM REQUIREMENTS

The system requirements for Documanage Bridge are composed of the system requirements for Documanage, IDS, your IDS client application, and the web server, if any.

Documanage Bridge is designed to provide high capacity and high performance if the system is configured with adequate resources.

The three major components of a bridge application — Documanage system, IDS server, and IDS client with web server — can, theoretically, all be on the same server machine. However, the performance and capacity of such a system would be limited, to say the least. Typically these three components are spread out over three or more separate high performance server machines on a robust network. See the Documanage documentation for more information on configuring Documanage for performance.

Typical strategies for boosting the performance of the bridge server are:

- Running more instances of IDS on a server machine
- Adding processor power, including additional processors, to the IDS server machines
- Installing redundant IDS server machines

See the IDS documentation for more information.

If your system does not provide the throughput that you need, you can use the tools mentioned in [Troubleshooting and Diagnostics on page 87](#) to identify the bottleneck component, and then take steps to increase that component's throughput.

Choosing the appropriate configuration for your bridge application can require careful study and planning. Contact Documanage Client Services for consultation.

Chapter 2

Installing the Bridge

This chapter tells you how to install and set up Docupresentment's Documanager Bridge.

You will find this information:

- [Introduction on page 8](#)
- [Components on page 9](#)
- [Requirements on page 10](#)
 - [Hardware on page 10](#)
 - [Software on page 10](#)
- [Installation on page 11](#)
 - [Graphical Installation on Windows and Unix on page 11](#)
 - [Terminal Installation on UNIX Operating Systems on page 18](#)
- [Uninstalling the Bridge on page 23](#)

INTRODUCTION

The Internet Document Server (IDS) provides application services, usually configured behind a Web server, to serve documents to users on an internet. Connecting to back-end document repositories like Documanager requires additional components called bridges. The Documanager Bridge provides the software rules, sample templates and configuration files which connect IDS users to a Documanager system.

The IDS documentation provides a complete overview of the Internet Document Server architecture. The Documanager Bridge includes both IDS server-side and client-side components. The bridge is a library of additional rules that are added to the IDS server directory (docserv directory) and are then configured with request-types using the IDS configuration file, docserv.ini. The Documanager API library is also installed to support this library. In the Windows installation, optional client-side asp (active server page) scripts provide sample pages that can be used as a starting point when creating a new Web site. Detailed information on the bridge rules is available in the Documanager Bridge Rules Reference provided with this product.

COMPONENTS

The software components that make up the Internet Document Server and its Bridges include:

- The IDS run time environment for your operating system
- Dynamically Shared Objects (DSOs)

NOTE: In Windows environments, DSOs are usually named **.dll*. In UNIX compatible environments, DSOs are usually named *lib*.so* or *lib*.sl*.

- Supporting configuration files, including docserv.xml and/or docserv.ini.
- Additions to the IDS configuration file, docserv.ini.

A Microsoft Windows installation also includes sample active server page (asp) scripts and Java server page (jsp) scripts that provide interfaces to most of the available bridge functions and are designed to work with the DmgSamples database.

REQUIREMENTS

The Documanager Bridge is compatible with the requirements of the Internet Document Server. It supports the same hardware and software platforms.

HARDWARE

Because of the additional software which comprises the Bridge, there is a nominal increase in the disk space required on the server and client machines to accommodate the installation process and the installed application and configuration files. Additional operational requirements may be adjusted in the Bridge configuration file, including the amount of disk space required to:

- Retrieve and convert documents from the Documanager archive,
- Cache recently retrieved and converted documents, and
- Retrieve and house resources needed to render print stream documents if you are storing and imaging Metacode or AFP print stream documents.

These disk space requirements vary depending on the size and file types used by your documents and the configured cache sizes, so specific requirements are not specified here.

Document translation and imaging are memory and CPU-intensive tasks. If you are going to use the IDS to perform these tasks, we recommend that you use fast processors and fast memory. Make sure that more than 512 MB of memory is available while the IDS server is running, but not processing documents.

Your Documanager system must have a network connection to the IDS/Bridge server.

SOFTWARE

To successfully run the UNIX installation scripts for IDS and the Bridge, you must have the following utilities available on the target system: awk, cpio, gunzip, sed, tar, uncompress, and uuencode. Make sure that they have been installed from your distribution packages before running the installation scripts.

To install and use the Documanager Bridge, you must have the base Internet Document Server software already installed on your server, including:

- The IDS run-time environment components
- Documaker Bridge/Shared Objects components —These components provide common document format translation services often used by bridge installations.

NOTE: All files on UNIX compatible operating systems must be in lowercase. You can use the up2low script utility to convert the case of file names you are moving from Windows to a UNIX compatible operating system. This utility is included on the Internet Document Server installation CD.

Also, for UNIX compatible operating system installations of IDS and CGI-BIN C-based clients, all directory names listed in the path to the installation directory and resources must consist of only lowercase letters.

Additional bridge requirements: Documanager 6.4 Service Release 1 or higher.

INSTALLATION

The Bridge can be installed to run under Windows, or it can be installed to run under one of three UNIX compatible (POSIX compliant) operating systems. These include Linux, Solaris, and AIX. The paragraphs that follow describe how to install the Bridge on both types of platforms.

GRAPHICAL INSTALLATION ON WINDOWS AND UNIX

With the Documanage Bridge version 3.4.7, both the Windows and Unix installations can be done via a graphical interface. You can download the Bridge installation via the Internet, or it can be supplied to you on a CD. The installation contains the server, client, bridges, resources, and documentation.

NOTE: Some graphics that follow may not match your installation exactly, i.e. color changes to the screens or minor release numbers in the titles. However, unless otherwise stated, the procedures remain the same for different releases.

To download the Bridge installation from the Internet, go to the Support web site at:

<http://www.skywiresoftware.com/NorthAmer/>

After you log in, click on Product Installation Downloads, then click Present. From there you can get the latest version of IDS and any bridges you need to run.

Preparing for the installation

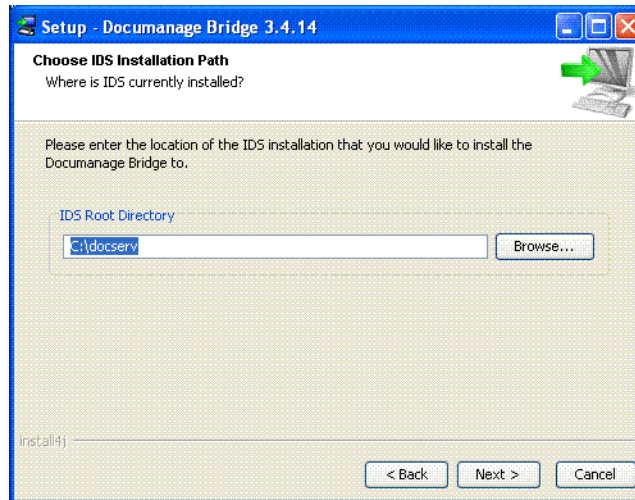
Before you start the installation...

- Be familiar with how your Web server is set up.
- Know the directory into which you are going to install the Bridge's components.
- Know the Address, Protocol, and Endpoint of the Documanage Router.
- Know the User name, Password, and Domain for the Bridge account.
- Know the source you are going to use for the user credentials.
- If you are going to use Trusted credentials, know the Name Variable, User Name and User Domain for the credentials.
- Be prepared to restart your computer after the installation is complete.
- Exit all programs currently running. If any programs are already installed *and in use* when you run the setup routine, conflicts will arise.

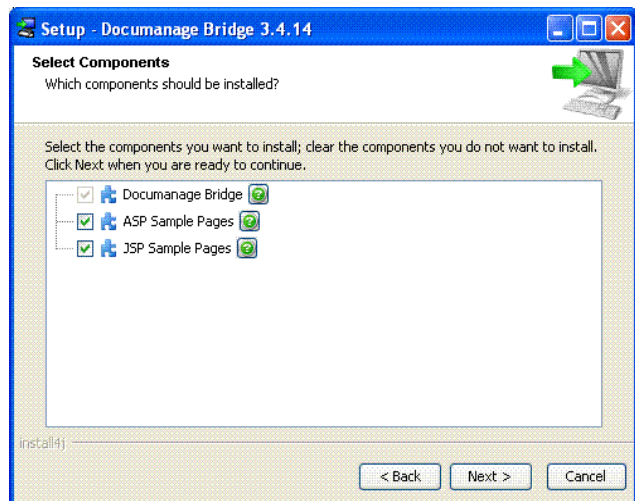
Installation steps

Follow these steps to install the Bridge:

- 1 Insert the Bridge installation CD in your CD drive or download the Documanage Bridge installation executable file from the Skywire Software web site into a folder on your hard drive.
- 2 Run the Bridge installation executable. Once the setup routine starts, follow the instructions which appear in the setup dialogs.
- 3 Select the location for the currently installed Internet Document Server.



- 4 If you select Custom on the Setup Type dialog, installation options appear in the Select Features dialog. Select one or more of the installation options. Click Next.

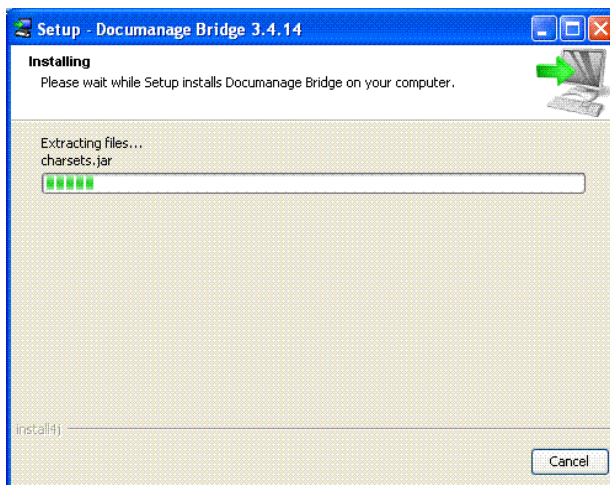


NOTE: Unless you are installing the Bridge on a server computer with a web server, you will usually install the Documanager Bridge and not the Sample ASP or JSP Templates. In this case, a single computer will run the IDS server and the client. A custom installation allows you to skip installing the sample templates if you are not using them or are performing a server-side only installation. The sample templates provide a good start towards developing your own application

Bridge 3.2 has a new rule and document properties that deal with document retention. These features are available only if the Bridge is connected to Documanager 6.4 SR4 or higher and if the Dmg_Api that the Bridge is using is version 1.4.1 or higher.

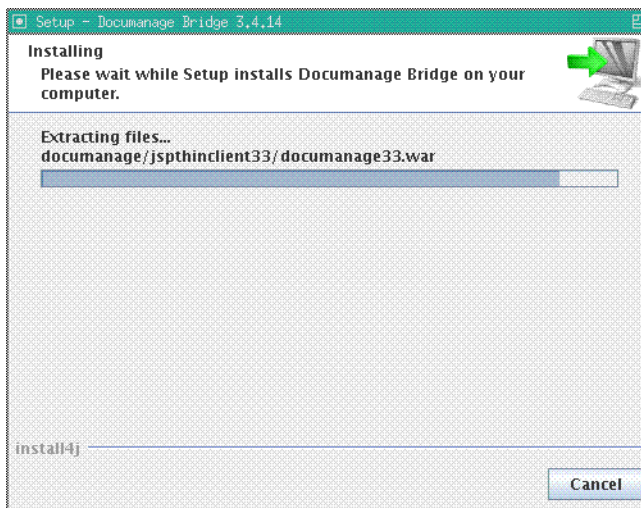
- 5 The Ready to Install the Program dialog appears. To change your settings, or to exit the installation, follow the instructions in the dialog. Click Install.

- 6 For a Windows installation, the Setup Status dialog appears with a progress bar that shows components are being installed. For a non-Windows installation, skip to Step 7.

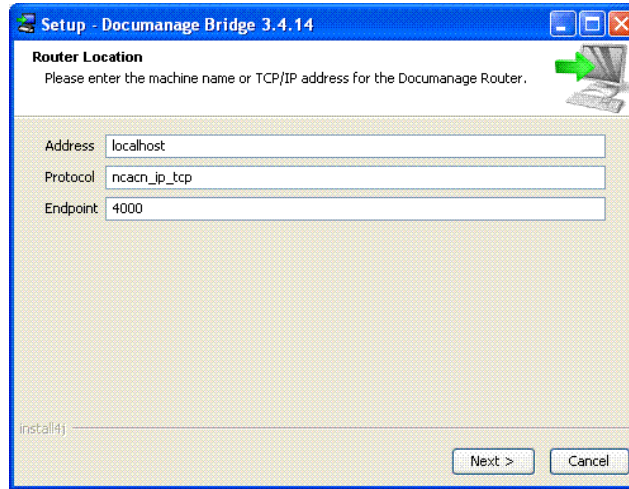


NOTE: If you install the Sample ASP Templates on a computer running Microsoft IIS Web Server, a “Virtual Directory” is created at the Default Web Site that provides a good start towards developing your own application.

- 7 The Setup Status dialog for a non-Windows (UNIX) installation appears with a progress bar that shows that components are being installed.

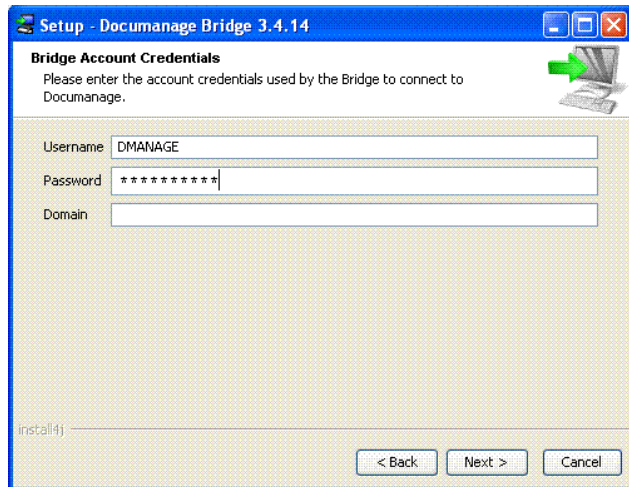


- 8 The Router Location dialog appears. Enter the machine name (network identity) or the TCP/IP address of the Documanage Router in the Address box. Click Next.

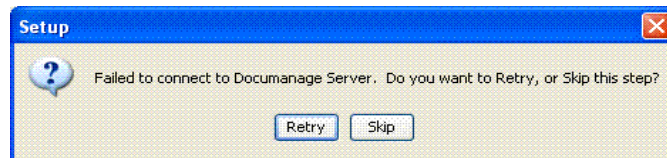


Refer to [Documanage] on page 66 for the definitions of the parameters in this dialog.

- 9 The Bridge Account Credentials dialog appears. Enter the User name, Password, and Domain of the Bridge Account. Click Next.

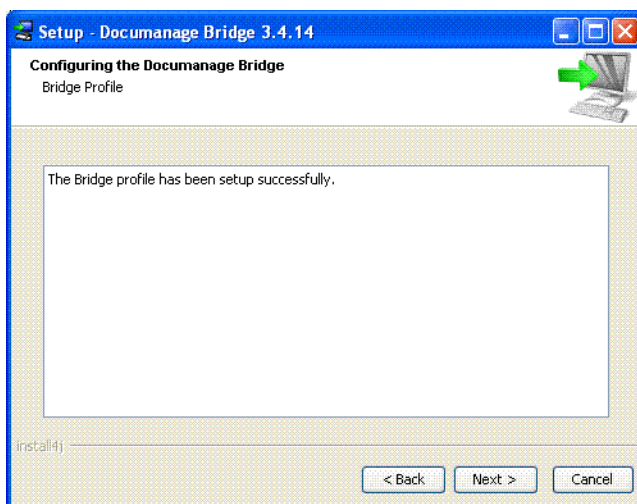


- 10 If the DmgBridge profile fails to log you into Documanage, a message appears that gives you a chance to enter different credentials.

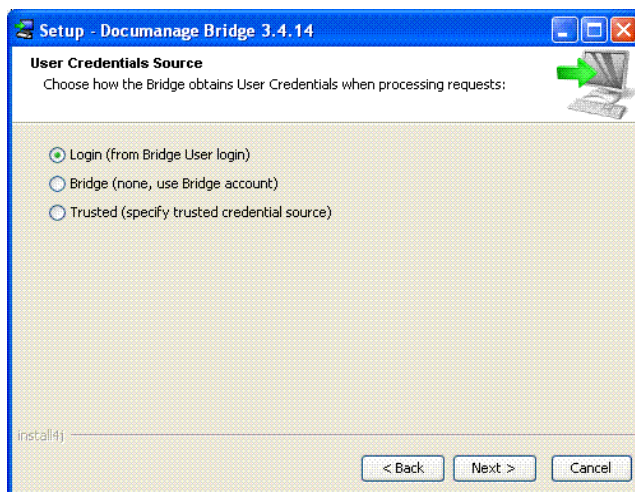


- Click Yes to make the Router Location dialog appear again, so you can modify your entries.

- Click No to allow the installation to continue without creating a DmgBridge profile. Without this profile, the Documanager Bridge will not be properly configured to connect to Documanager. See the Dmgprofile Utility document (dmgprofile.pdf) to manually create a profile for your Documanager Bridge after installation completes.
- The Bridge Profile dialog appears if the profile has been set up successfully.



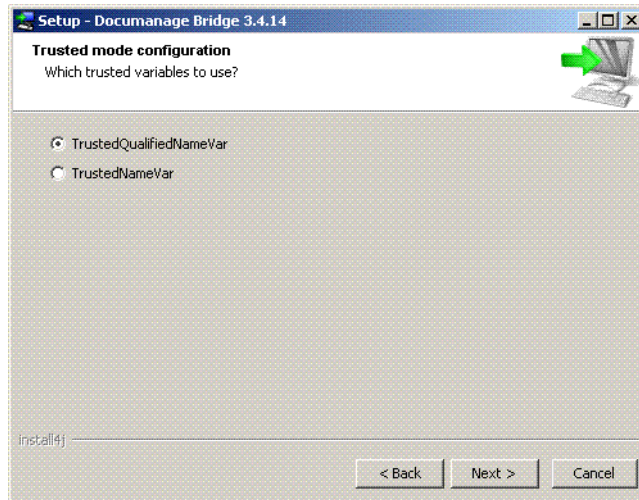
- 11 The User Credentials Source dialog appears. Select the Login, Bridge, or Trusted option. Click Next.



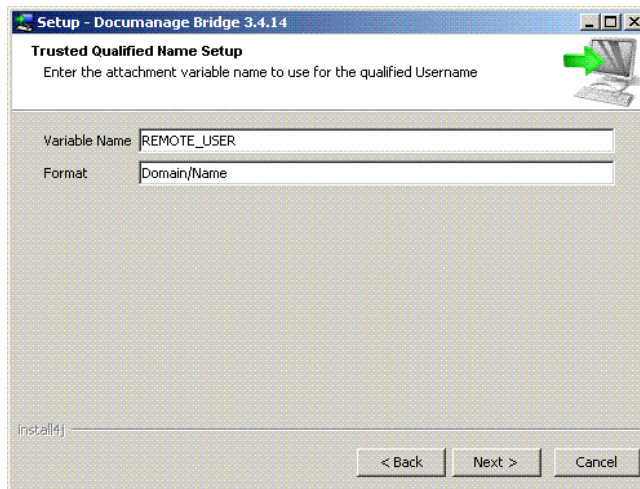
Refer to [\[UserCredentials\]](#) on page 67 for descriptions of these options. Item 1 describes the Login option, Item 3 describes the Bridge option, and Item 2 describes the Trusted option.

- 12 If you have selected the Login or Bridge option, the InstallShield Wizard Complete dialog appears. Go to step 16.

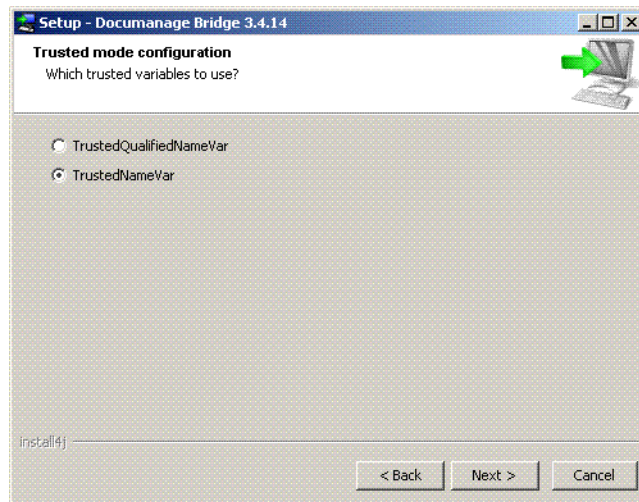
- 13 If you have selected the Trusted option, the Trusted Mode Configuration appears. Select the configuration mode to use for Trusted Credentials. If using TrustedQualifiedNameVar, go to Step 14. If using TrustedNameVar, go to Step 15. Click Next.



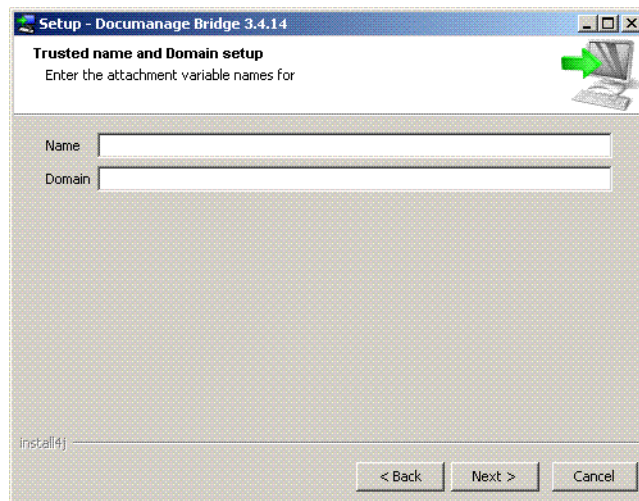
- 14 If the TrustedQualifiedNameVar is selected, the next dialog will ask you to supply the Attachment Variable name to use for the qualified Username. Click Next to complete the installation (Step 17).



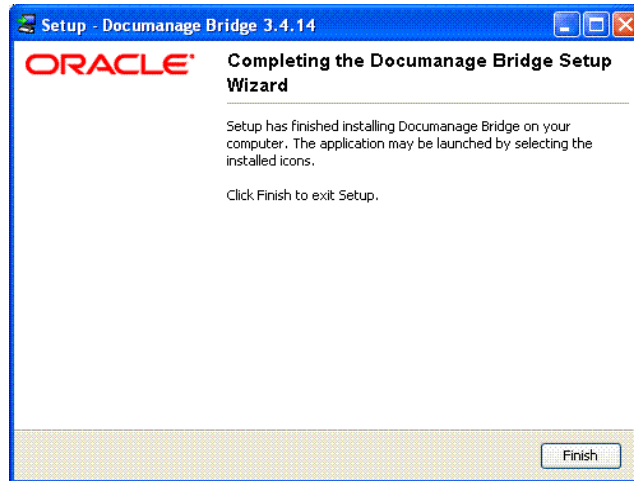
- 15 If needed, select the TrustedNameVar option in the Trusted Mode Configuration dialog.



- 16 Next, enter the variable names for the TrustedNameVar. Click Next to complete the installation.



- 17 The InstallShield Wizard Complete dialog appears. Follow the instructions in the dialog to select the Yes or No option, if required. Click Finish.



NOTE: The InstallShield Wizard Complete dialog may not ask if you want to restart your computer. Whether this is required or not depends on your operating system and the files that you may already have installed or in use at the time of installation.

TERMINAL INSTALLATION ON UNIX OPERATING SYSTEMS

Each UNIX compatible installation consists of a single shell script for the appropriate UNIX compatible platform. The script installs the Bridge and helps you configure it to run in your operating environment. It contains all of the bridge software, and it runs the installation steps and basic configuration steps.

Script names The script name for each type of UNIX compatible installation differs depending on the operating system supported. The script names follow this pattern:

`dmgbridge_PPP_V_V_V.sh`

where:

PPP	Represents the platform for which the script is intended: aix = IBM AIX lnx = Linux win32 = Windows
V_V_V	Represents the version release of the bridge and installation script (major, minor, and build versions)

Examples Here are some examples:

`dmgbridge_aix_3_4_14.sh`

This installs Documanage Bridge version 3.4 on AIX.

`dmgbridge_linux_3_4_14.sh`

This installs Documanage Bridge version 3.4 on Linux.

Loading the scripts You can download the Bridge installation via the Internet, or you can load it onto your machine from a CD.

To download the Bridge installation script via the Internet:

- Go to the Support web site at:
<http://www.skywiresoftware.com/NorthAmer/>
- After you log in, click on *Product Installation Downloads*, then click *Present*. From there you can download the latest version of IDS and any bridges you need to run.

To load the installation script onto your machine from a CD:

- Insert the Documanager Bridge CD in a workstation and copy the installation files from the CD over to the UNIX compatible operating system server using NFS, FTP, SaMBa, SFTP, or another file transfer method.

This requires an account, a file transfer method, and web server pre-configuration but not administrator assistance or direct server access. Make sure the copied setup*. * file has executable permissions set to allow it to execute as a script file.

- Insert the Documanager Bridge CD in your UNIX compatible operating system server (mount the CD drive device if it is not auto-mounted) and run the installation directly from the CD.

This requires direct server access, mount permissions, and administrator assistance.

NOTE: All files and directory names listed in the path to the installation directory on UNIX compatible installations must be in lowercase. You may also need to change the permissions of the setup file to executable when you copy them from one location to another. Use the `chmod u+x filename` command to do this.

Using the Installation Script

To install the Bridge on a UNIX compatible system, start its `dmgbridge_PPP_V_V_V.sh` file from a terminal using an xterm, console, or telnet session and then follow these steps to install the Documanager Bridge:

- 1 Insert the Documanager Bridge installation CD in your CD drive or download the Documanager Bridge installation executable file from the Oracle web site into a folder on your hard drive.
- 2 To run the Documanager Bridge installation executable in console or terminal mode add `-c` after the installation executable. Here is an example:

```
$ ./dmgbridge_aix_3_4_14.sh -c
```

- 3 Once the setup routine starts, follow the instructions which appear on the screen. Type `o`, or press ENTER to start the installation. Type `c` to cancel the installation.

```
Unpacking JRE ...
Starting Installer ...
This will install Documanager Bridge on your computer.
OK [o, Enter], Cancel [c]
```

- 4 Select the location for the currently installed Internet Document Server.

```
Where is IDS currently installed?  
[/home/lellison/docupresentment/docserv/dmgbridge]  
/home/lellison/docupresentment/docserv  
Checking for previous IDS Installations...
```

- 5 If you select Custom on the Setup Type window, installation options appear in the Select Features window. Select one or more of the installation options. Click Next.

```
Which components should be installed?  
1: Documanager Bridge  
2: JSP Sample Pages  
Please enter a comma-separated list of the selected values or [Enter]  
for the default selection:
```

```
[1,2]  
1,2
```

The installation routine lists the files as it installs them.

```
Extracting files...  
Downloading ...  
Extracting files...  
...  
.install4j/  
.install4j/uninstall.png  
uninstall  
/  
cache/  
dmgprofile  
dmgprofile.sh  
dmg_brc.ini  
dmg_brmime.ini  
dmg_brs.ini  
dmg_reqtypes.ini  
libdmg_api.so  
libdmg_br.so  
/  
documanager/  
documanager/jsptthinclient33/  
documanager/jsptthinclient33/documanager33.war
```

The Router Location Setup window appears.

- 6 Enter the machine name (network identity) or the TCP/IP address of the Documanager Router in the Address box.

```
Please enter the machine name or TCP/IP address for the Documanager  
Router.
```

```
Address  
[localhost]  
10.143.215.206  
Protocol  
[ncacn_ip_tcp]  
ncacn_ip_tcp  
Endpoint  
[4000]
```

The Bridge Account Credentials setup appears.

- 7 Enter the user name, password, and domain of the bridge account. Note: the password appears in console mode.

```
Please enter the account credentials used by the Bridge to connect
to Documanager.
```

```
Username
[]
dmgbridge
Password
bridgePass
Domain
[]
ORACLE
```

If the DmgBridge profile fails to log you into Documanager, the installation routine lets you try again with different credentials.

```
Failed to connect to Documanager Server. Do you want to Retry, or
Skip this step?
```

```
Retry [1], Skip [2]
```

```
2
```

Enter **1** to make the Router Location Setup window appear again so you can modify your entries.

Enter **2** to allow the installation to continue without creating a DmgBridge profile. Without this profile, the Documanager Bridge will not be properly configured to connect to Documanager. See the Dmgprofile Utility document (dmgprofile.pdf) to manually create a profile for your Documanager Bridge after installation completes.

The Bridge Profile Setup window appears if the profile has been set up successfully.

```
Bridge Profile
```

```
The Bridge profile has been setup successfully.
```

```
[Enter]
```

The User Credentials Source window appears.

- 8 Select the Login, Bridge, or Trusted option. Refer to [\[UserCredentials\] on page 67](#) for descriptions of these options. Item 1 describes the Login option, Item 3 describes the Bridge option, and Item 2 describes the Trusted option.

```
Choose how the Bridge obtains User Credentials when processing
requests:
```

```
Login (from Bridge User login) [1, Enter], Bridge (none, use Bridge
account) [2], Trusted (specify trusted credential source) [3]
```

If you have selected the Login or Bridge option, the InstallShield Wizard Complete Setup window appears. Go to step 12.

If you have selected the Trusted option, the Trusted Mode Configuration window appears. Select the configuration mode to use for Trusted Credentials. If you are using TrustedQualifiedNameVar, go to Step 10. If you are using TrustedNameVar, go to Step 12.

```
Which trusted variables to use?
```

```
TrustedQualifiedNameVar [1, Enter], TrustedNameVar [2]
```

- 9 If you select `TrustedQualifiedNameVar`, the next window asks you to supply the attachment variable name to use for the qualified user name. Go to Step 12.

```
Enter the attachment variable name to use for the qualified Username  
Variable Name  
[REMOTE_USER]
```

```
Format  
[Domain/Name]
```

- 10 If needed, select the `TrustedNameVar` option in the Trusted Mode Configuration setup.
- 11 Enter the variable names for the `TrustedNameVar`. Go to Step 12.

```
Enter the attachment variable names for  
Name  
[]  
LDAP_NAME  
Domain  
[]  
LDAP_DOMAIN
```

The InstallShield Wizard Complete window appears.

- 12 Follow the instructions in the window to select the Yes or No option, if required. Click Finish.

```
processing  
Finishing installation...
```

UNINSTALLING THE BRIDGE

You uninstall Documanager Bridge differently, depending on whether it has been installed in a Windows or a UNIX compatible environment.

UNINSTALLATION FROM WINDOWS

In Windows, use the Control Panel | Add or Remove Programs option to remove Documanager Bridge. Data files saved after you installed Documanager Bridge are preserved, as are registry entries.

If necessary, you must remove the Microsoft components separately.

UNINSTALLATION FROM UNIX

To uninstall Internet Document Server and Documanager Bridge from a UNIX-compatible operating system, remove the base installation directory and all its sub-directories. It may also be necessary to remove any HTML and CGI-BIN directories if they are outside the base installation directory.

Chapter 3

Creating a Bridge Application

A *bridge application* consists of the customizations that adapt the Documanager Bridge for use in a particular application. These customizations include:

- Bridge server rules lists in docserv.ini to handle all of the transactions the Bridge will be expected to perform.
- A bridge (IDS) client application that handles the inputs and outputs for the rules lists configured in docserv.ini.
- Any HTML templates or other such files needed to support the client application.
- Settings in the bridge configuration files that are appropriate for the application.

As someone with the skills to install, configure, and manage IDS and to write IDS client applications, you will already be familiar with the main concepts and techniques involved in writing a bridge application. This section discusses aspects of writing a bridge application that may be unique to Documanager Bridge.

Documanager Bridge puts no restrictions on the client application environment (CGI, ASP, JSP, and so on) or on the choice of queues. Any environment and any queues that can be used with IDS can be used with Documanager Bridge.

The Documanager Bridge provides a few features to help you with certain environments. If you are using the IDS standard CGI client, there are some rules that you can use on rules lists in the standard client's configuration file, docclnt.ini. The names of those rules begin with "DmgBrc..." or "DmgBru...". See the documentation of those rules for details.

If you are using ASP, the ASP cookie collection requires the cookie expiration date to be in a different format than that used in other environments. Since the Documanager Bridge rule `DmgBrsValidateSession` provides a cookie expiration date for you, the format of that date is settable so that it can be used with ASP. See the documentation of `DmgBrsValidateSession` for details.

Many but not all INI options are mentioned in this section. See the section “Configuring the Documanager Bridge” for a complete description of the available configuration options.

SERVER RULES LISTS

One of the first tasks in creating a Documange Bridge application is to decide what transactions will be needed and to create server rules lists to handle those transactions.

Here is a possible server rules list from docserv.ini that runs bridge rules. The names of bridge server rules begin with *DmgBrs*. The main purpose of the transaction is to search for documents, so this rules list runs the rule called *DmgBrsQueryDocuments*. We have decided to set the *ReqType* to *QUERY_DOCS* in our example, although you could choose any *ReqType* suitable for your application.

```
[ ReqType:QUERY_DOCS ]
  function = atcw32->ATCLogTransaction
  function = atcw32->ATCLoadAttachment
  function = atcw32->ATCUnloadAttachment
  function = dmg_brw32->DmgBrsCopyAttachment
  function = dmg_brw32->DmgBrsValidateSession
  function = dmg_brw32->DmgBrsEncodeURLField,HITS*.FLD_KEYSTRING
  function = dmg_brw32->DmgBrsQueryDocuments
```

ATCLogTransaction, *ATCLoadAttachment*, and *ATCUnloadAttachment* are standard IDS rules. Note that *ATCUnloadAttachment* executes on the reverse run of the rules list, so it is positioned to run after the forward and reverse runs of all bridge rules.

The rules *ATCLogTransaction* through *DmgBrsValidateSession* will appear on nearly every bridge server rules list.

DmgBrsCopyAttachment copies all of the attachment variables from the input queue to the output queue, where the remaining bridge rules expect to find them. Since all remaining bridge rules depend on this, it is positioned to run before the forward run of any other bridge rules. Use it on all bridge server rules lists except where noted in the documentation of a particular rule.

DmgBrsValidateSession is positioned to run before the forward run of *DmgBrsQueryDocuments* because *DmgBrsQueryDocuments*, like nearly all bridge server rules, checks for a valid session as indicated by the outputs of *DmgBrsValidateSession*. Use *DmgBrsValidateSession* this way on all bridge server rules lists except where noted in the documentation of a particular rule.

DmgBrsQueryDocuments is the rule that carries out the purpose of this transaction. It executes a query and returns a list of documents. The position it occupies is typical for the placement of the rules that are unique to the request. The rule *DmgBrsEncodeURLField* appears on the rules list to URL encode some of the outputs of *DmgBrsQueryDocuments* so they can be used in URLs on a web page.

These IDS rules must be configured in particular ways with Documange Bridge:

- See *DmgBrsInit* for instructions on configuring the INI rules list.
- See *DmgBrsIdle* for instructions on configuring the SAR rules list.
- See *DmgBrsCheckPausedServer* for instructions on configuring the PAUSE rules list.

CREATING WEB PAGES

One of the chief responsibilities of your bridge application is to create web pages that allow the user of the system sitting at a browser to request information or files from Documanager and that display the results of the request.

There are two basic types of request that can be sent to the Documanager Bridge:

- Request for information
- Request for a file

The HTML input tag is typically used to allow a user to input the elements of a request for information. An HTML anchor tag with an HREF field is often used when a file is to be downloaded, as in the case of a document file that the user may want to retrieve and use. An HTML Img tag is often used when an image file is to be downloaded for display on the web page, as in the case of a thumbnail image.

For any of these tags, the web address that identifies your bridge application, known as the bridge URL, is required. You may hard code the bridge URL, but you may find it easier to use the variable BRIDGEURL, returned by DmgBrsValidateSession.

A crucial element of all requests is, of course, the ReqType. You must pass the correct ReqType with all of your requests.

Many of the elements of the query string for a request are available in the variables that resulted from the previous request.

For some examples of the use of attachment variables with HTML Anchor and Img tags, see DmgBrsPresentDocumentPage and DmgBrsQuery.

See the sample bridge applications for examples of using bridge attachment variables in web pages.

URL ENCODING

The values of any attachment variables that are being returned from a bridge rule must, of course, be URL encoded if you plan to use them in the HREF field of an HTML anchor tag. See the description of the rule `DmgBrsEncodeURLField` for more information and examples.

BRIDGE ENCODING

Many Documanager Bridge server rules require that property names used in particular attachment variable names or in INI files be *bridge encoded*.

Bridge encoding provides a way to use the names of properties that may have certain characters in them in places where those characters could cause problems. The encoding, which is similar in concept to URL encoding, replaces those characters with substitute characters that do not cause problems. The descriptions of specific rules specify which property names must be bridge encoded in which situations.

For example, property names may include the bracket characters '[' and ']'. If such a name were to be used in an INI file without encoding the brackets, it could cause confusion because the brackets would look like INI section delimiters.

A Documanager Bridge application never has to carry out any actual bridge encoding or decoding. Instead, the application can just use the bridge encoded version of the property name, as returned by the rule DmgBrsLogPropertySets. See DmgBrsLogPropertySets for more information.

In bridge encoding, a potentially problematic character is replaced by the character '\$' followed by two digits which are the hexadecimal representation of the character. Since '\$' is the escape character for bridge encoding, it is one of the characters that must be encoded. The following table lists the characters that are bridge encoded, and what their encodings are.

Character	Bridge Encoding
\$	\$24
the space character	\$20
[\$5B
]	\$5D

UI CONFIGS

With IDS and the standard CGI client, you can create multiple user interfaces (UIs) that are designed to meet the needs of different groups of users. Each UI configuration has its own name and a directory path associated with it. The standard CGI client uses the configuration name to find the associated directory path, and uses this directory path when looking for HTML templates to render pages. By defining multiple configuration names, multiple UIs can be created and accessed from the Internet. Documanager Bridge fully supports this IDS functionality.

The bridge session cookie is used to preserve the UI CONFIG between transactions for a given user. The default UI CONFIG is set with the INI option [ValidateSession] DefaultUIConfig. If this INI option is set, the rule DmgBrsValidateSession will output the correct value of the CONFIG variable, as stored in the cookie or amended in the current transaction, for use in your client application. If this INI option is not set, DmgBrsValidateSession will assume that you are not using UI CONFIGs in your Documanager Bridge application, and will not output CONFIG.

To learn how to set CONFIG, change the CONFIG for the current and subsequent transactions, set a TEMP_CONFIG for the current transaction only, and reset the CONFIG to the default, see DmgBrsValidateSession and DmgBrsLogin.

NOTE: Although the configuration name is stored in the cookie, the configuration value is not security protected. Users can manually add **&config=x** to the end of any URL to change their UI CONFIG. Multiple UI configurations cannot be used to provide security or hide data from particular users.

USER SECURITY

Documanager Bridge provides these sources of authentication of individual users:

- *Login*, which uses the rule DmgBrsLogin.
- *Trusted*, which provides for authentication of the user name and domain by some agent outside of Documanager Bridge, such as the web server.
- *Bridge*, which gives everyone the same access without the need for authentication.

See the INI options for the UserCredentials control group for a detailed explanation of these three sources of authentication and how to configure them. See DmgBrsLogin for more information on using Login. See DmgBrsValidateSession for more information on how the user's session is maintained and validated.

If you use Login, then you must use Documanager Bridge cookies. This means that you must set the INI option [ValidateSession] UseCookies to ON.

LOGIN

If you use Login, then you will need to create a web page to solicit the user's credentials. This is referred to as a Login UI display page.

The page will be used in two situations: initial login, and an Ad Hoc Login (see the section "Ad Hoc Logins"). The page must run the rules list configured in the INI option [ReqTypes] LoginExecute, which must run DmgBrsLogin. It must submit the required input variables for the rule DmgBrsLogin and, to support Ad Hoc Logins, must also submit the variable BRS_DESTINATION, passing through whatever BRS_DESTINATION is passed to it. The page may also display the variables output by DmgBrsValidateSession whose names begin with BRSLOGIN_, which signal the reason for an ad hoc login, if they exist.

For initial login, create a ReqType for initial login and configure the INI option [ReqTypes] LoginUIDisplay to that ReqType. Create a rules list that runs DmgBrsCopyAttachment and DmgBrsValidateSession, like the one immediately below. No other Bridge rules are needed on this list. DmgBrsValidateSession will recognize the rules list as that for the LoginUIDisplay and will set BRS_DESTINATION accordingly. DmgBrsValidateSession will also make BRIDGEURL available to the Login UI Display page. For the example below, we would set the INI option [ReqTypes] LoginUIDisplay to LOGIN_UI.

```
[ ReqType:LOGIN_UI ]
function = atcw32->ATCLogTransaction
function = atcw32->ATCLoadAttachment
function = atcw32->ATCUnloadAttachment
function = dmg_brw32->DmgBrsCopyAttachment
function = dmg_brw32->DmgBrsValidateSession
```

The ResType for LOGIN_UI will then handle both the result of calling ReqType LOGIN_UI, and the result of an Ad Hoc Login, when DmgBrsValidateSession has changed the ReqType to LOGIN_UI

BridgeURLs and Secure Login

Since the Login UI Display page must submit the user's password over a network, possibly the internet, we recommend using SSL (Secure Socket Layer) for your bridge application. If you do not want to use SSL for your entire bridge application, you can configure the bridge to use SSL for login only. See the INI options [BridgeURL] for more information on configuring the proper bridge URLs for SSL or SSL for login only.

Automatic Logout when Quitting the Browser

With the INI option [ValidateSession] CookiesPersist, you can configure DmgBrsValidateSession to signal the bridge application to create a bridge cookie without an expiration date. Whenever the user quits the browser, the browser discards cookies that have no expiration date.

Since the fact that the user has logged in is recorded in the cookie, discarding the cookie has the effect of logging the user out. The user will have to log in the next time they start their browser and attempt to access the bridge. If they do not initially go to the Login UI Display page, DmgBrsValidateSession will detect the lack of the cookie and will initiate an Ad Hoc Login.

THE BRIDGE COOKIE

The primary purpose of the bridge cookie is to maintain the user name and domain (validated by DmgBrsLogin) between transactions when the INI option [UserCredentials] Source is Login. This has the effect of maintaining a virtual session for the user even though the user is not actually logged into Documanager. The cookie also contains other data that is used to validate the user session, such as the time of the last transaction, the user's IP address, and so on. When Source is Login, cookies must be turned on with the INI option [ValidateSession] UseCookies, or the Bridge will not initialize.

Other purposes of the Bridge Cookie are to maintain the UI CONFIG setting, if any, and the User Data, if any, for a particular user between transactions. If the INI option [UserCredentials] Source is not Login, you may turn cookies off. If, however, cookies are off, the rule DmgBrsValidateSession will have no way to maintain the UI CONFIG and the User Data for you between transactions. If they are needed in your Bridge Application, then the application will have to maintain them. See DmgBrsValidateSession for more information about User Data.

The name of the bridge cookie is *DMGBRSESS*, for *Documanager Bridge Session*. This name may change without notice. Get the bridge cookie name from the value of the variable HTTP_COOKIE_NAME from the reverse run of DmgBrsValidateSession.

Since the cookie contains security-related information, it is scrambled unless you temporarily turn scrambling off for testing purposes (using [ValidateSession] ScrambleCookies). No password is stored in the bridge cookie.

Configure the cookie with INI options in [ValidateSession]. Also see the rule DmgBrsValidateSession.

When cookies are in use, it is the responsibility of your bridge application to transmit the cookie to the browser. The outputs of the reverse run of the rule DmgBrsValidateSession provide all of the information necessary to create the cookie, including its name, its content, and its expiration date, if any. There are two situations to handle: downloading a web page and downloading a file.

The way you create a cookie will, of course, depend on the environment in which you are creating the Documanager Bridge client application. In ASP, for example, you will use the cookie collection. If you use the Standard CGI Client and HTML templates, when downloading a web page with the rule IRCUnloadPage, you can configure an HTTPHeader template similar to the following example.

```
Status: #HTTP_STATUS,%s#
Set-cookie: #HTTP_COOKIE_NAME,%s#=#HTTP_COOKIE_TEXT,%s#; path=/;
expires=#HTTP_COOKIE_DATE,%s#
Content-Type: text/html
```

In the above example, you can remove Set-cookie entirely if cookies are not in use in your application. You can remove expires= if [ValidateSession] CookiesPersist is OFF, or alternatively you could add conditional processing directives to the template to detect this condition and output the Set-Cookie line without an expiration date. The rule DmgBrsCopyAttachment, which is used on virtually all bridge transactions, outputs the variable HTTP_STATUS, which may be changed by other bridge rules as necessary.

For more information on handling the cookie when downloading a file, see [Downloading Files from the Client Application on page 49](#).

REQTYPES

In Documanager Bridge, you can use any ReqType you like for the transactions you create. In most cases, bridge rules will leave the Reqtype unchanged, and you will handle it as the ResType.

There are a few cases where Documanager Bridge needs to know about the ReqTypes you have chosen so that it can detect what kind of transaction is occurring, or so that it can change the ReqType appropriately to signal certain conditions. Use the INI options in the ReqTypes control group to configure those ReqTypes.

For those cases in which bridge server rules change the ReqType, your bridge application must detect that this change has happened, and must be prepared to handle the change as the ResType of the transaction.

- Some rules change the ReqType to that configured in the INI option [ReqTypes] Redirect. See the sections on Redirection, Ad Hoc Login, and Bridge Temp Files.
- Some rules change the ReqType to that configured in the INI option [ReqTypes] LoginUIDisplay. See the sections on User Security and Ad Hoc Login.
- Most rules change the ReqType to that configured in the INI option [ReqTypes] Failure in order to signal an error. See the section “How Server Rules Signal An Error”.

REDIRECTION

In certain circumstances, Documanager Bridge server rules change the ReqType to that configured in the INI option [ReqTypes] Redirect. This means that the Documanager Bridge server rule has determined what the next transaction must be. For example, DmgBrsLogin will automatically redirect to the transaction that should occur next.

It is the responsibility of the bridge application to correctly handle the redirection as a ResType. Whenever a Documanager Bridge server rule changes the ReqType to [ReqTypes] Redirect, it will also output the variable BRS_DESTINATION, which will contain the URL that leads to the next transaction.

Here is an example of an HTML template that could be used with the Standard CGI Client to handle redirection.

```
<HTML>
  <HEAD>
    <META HTTP-EQUIV="Content-Type" content="text/html; charset=iso-
8859-1">
    <TITLE>Processing...</TITLE>
  </HEAD>
  <BODY>
    <META HTTP-EQUIV="Refresh" CONTENT="0;URL=#BRS_DESTINATION,#">
  </BODY>
</HTML>
```

In the above, notice that the META tag has been used to refresh the contents of the page immediately with the URL in BRS_DESTINATION.

AD HOC LOGINS

If the INI option [UserCredentials] Source is set to Login, the rule DmgBrsValidateSession, which will appear on almost all server rules lists, may signal an ad hoc login by changing the reqtype to that configured in the INI option [ReqTypes] LoginUIDisplay or [ReqTypes] Redirect. In your bridge application, you must be sure to check for this condition and handle it appropriately.

See DmgBrsValidateSession for more information on ad hoc logins.

HOW SERVER RULES SIGNAL AN ERROR

To signal an error, a server rule will change the ReqType to the Failure reqtype, as configured in the INI option [ReqTypes] Failure. In your bridge application, you must always be sure to check for this condition and handle it appropriately.

In addition to setting the ReqType to the Failure reqtype, a rule that has failed will set the following output attachment RecordSet so that information about the error can be displayed to the user if desired.

Variable	Description
BRS_FAILURE	Contains the number of errors, typically 1.
BRS_FAILUREx.CODE	x is the record number of this field. An error code associated with the error.
BRS_FAILUREx.MSG	x is the record number of this field. A message associated with the error.

If a rule fails, do not expect to find its output attachment variables.

DATES AS INPUTS AND OUTPUTS

When Documaker Bridge server rules output the value of a Property that is of type `DateTime`, the value will be a string in the format `yyyy-mm-dd hh:mm:ss`. For example, “2004-01-12 23:00:00” is 12 January 2004 at exactly 11 PM. If the type is `Date` or `Time`, the irrelevant portion of the string is omitted.

When values are required as inputs for Properties of type `Date`, `Time`, or `DateTime`, a variety of formats are acceptable.

For the date portion of the string for `Date` and `DateTime` Properties, the recognized delimiters when all parts are passed in as numeric values are ‘/’ or ‘-’, and the order of the values may be `m/d/y` or `y/m/d`. The year must be four digits or it cannot be recognized. Month and day may be one or two digits. The time portion may be omitted from a `DateTime`.

For example, April 15, 1951 could be “4/15/1951”, “04-15-1951”, “4-15-1951”, “1951-4-15”, “1951/04/15”

For the string for `Date` Properties, the month may alternatively be spelled out, or the first three letters of the month may be used. Formats can be “spelledmonth d, y” or “d spelledmonth y”.

For example, April 15, 1951 could be “Apr 15, 1951”, “15 April 1951”, “April 15, 1951”, “15 Apr 1951”

For the time portion of the string for `Time` and `DateTime` Properties, the recognized delimiters are ‘:’, ‘.’, or ‘-’, and the order of the values must always be `h:m:s`. Hours are 0 through 23. AM and PM are not recognized.

For example, 11:59:22 PM could be “23:59:22”, “23.59.22”, or “23-59-22”.

NULL VALUES AS INPUTS AND OUTPUTS

Sometimes the database fields underlying Extended Folder, Project, and Document Properties are configured in the database to accept SQL NULL values. If that is the case, you may want to configure the bridge to distinguish between NULL values and empty (for strings) or 0 (for numeric) values when the values of these Extended Properties are being input or output in attachment variables. To do so, use the INI options in the section [NullValues] in dmgs_brs.ini.

See the description of the INI section [NullValues] in [Configuring the Documanager Bridge on page 63](#), the operators IS NULL and IS NOT NULL in [SQL Builder Syntax on page 41](#), and the output attachment variables of the rule DmgBrsListCabinets, described in the Rules Reference.

SQL BUILDER SYNTAX

Bridge server rules accept SQL filters and orderby clauses as inputs. For example, the rule DmgBrsQueryDocuments allows filtering by Folder or Document properties, and ordering results by Folder or Document. You may construct these filters and pass them directly to the rules in the specified variables. Or, you can use SQL Builder Syntax, provide property names and values, and operators where required, and let the Bridge rule construct the filter or orderby clause for you.

The SQL Builder syntax available for filters includes:

- Extended Folder Property Filter, to replace the FLD_FILTER variable
- Extended Document Property Filter, to replace the XDOC_FILTER variable
- Extended Project Property Filter, to replace the XPRJ_FILTER variable.
- Basic Document Property Filter, to replace the DOC_FILTER variable
- Basic Diary Property Filter, to replace the DIARY_FILTER variable

The SQL Builder syntax available for order by includes:

- Folder OrderBy, to replace the FLD_ORDERBY variable
- Document OrderBy, to replace the DOC_ORDERBY variable
- Diary OrderBy, to replace the DIARY_ORDERBY variable
- Project OrderBy, to replace the PRJ_ORDERBY variable

FILTER SYNTAX

The names of all variables for building filters begin with QUERY_. You must pass in a variable or variables to specify the value or values of a Property to be included in the filter, and a variable to specify the *FilterOperator* to be applied to the Property.

The list of available *FilterOperators* appears after the description of the filter syntax. Some operators apply only to certain Property types, so it is important to be aware of the type of the Property. *FilterOperator* is case-insensitive.

Empty Property values are not included in the filter.

Criteria for multiple Properties are ANDed together.

Building extended folder
property filters

```
QUERY_XFLD_Lx_VAL_FolderPropertyName = PropertyValue
QUERY_XFLD_Lx_VAL2_FolderPropertyName = PropertyValue
QUERY_XFLD_Lx_OP_FolderPropertyName = FilterOperator
```

“XFLD” indicates “Extended Folder”. “VAL” is “value”, and “OP” is the operator that goes with the corresponding Property name. The x is the one-based Cabinet level and *FolderPropertyName* is the Bridge-encoded Folder Property Name, as it appears in the Documanager Administrator for the business data table that is mapped to this level of the intended Cabinet. Note that *FolderPropertyName* is case-sensitive, and it is not the DB Field Name. The name of the Cabinet corresponding to these Extended Folder Properties is supplied in a separate variable (CABINET) for all the rules that use these filters. BETWEEN is the only Operator that requires VAL2. VAL2 should not be used for any other operator.

This example illustrates the use of the IN operator, and supplying criteria for more than one Property.

```
QUERY_XFLD_L1_VAL_Last$20Name = Smith
QUERY_XFLD_L1_OP_Last$20Name = equals
QUERY_XFLD_L1_VAL_First$20Name = 'Joe', 'Sally', 'Chris'
QUERY_XFLD_L1_OP_First$20Name = in
```

The above will cause a filter to be created that will find Folders whose Last Name Property is Smith AND whose First Name Property is Joe, Sally, or Chris.

This example illustrates the use of the BETWEEN operator.

```
QUERY_XFLD_L1_VAL_Start$20Date = 10/15/02
QUERY_XFLD_L1_VAL2_Start$20Date = 10/31/02
QUERY_XFLD_L1_OP_Start$20Date = between
```

Building extended project
property filters

```
QUERY_XPRJ_VAL_ProjectPropertyName = PropertyValue
QUERY_XPRJ_VAL2_ProjectPropertyName = PropertyValue
QUERY_XPRJ_OP_ProjectPropertyName = FilterOperator
```

“XPRJ” indicates “Extended Project”. “VAL” is “value”, and “OP” is the operator that goes with the corresponding Property name. *ProjectPropertyName* is the Bridge-encoded Project Property Name, as it appears in the Documanager Administrator for the business data table that is mapped to the intended Cabinet. Note that *ProjectPropertyName* is case-sensitive, and it is not the DB Field Name. The name of the Cabinet corresponding to these Extended Project Properties is supplied in a separate variable (CABINET) for all the rules that use these filters. BETWEEN is the only Operator that requires VAL2. VAL2 should not be used for any other operator. The use of this syntax is identical to the use of the syntax for building Extended Folder Property filters except for the Cabinet Level designator required when building Folder Property filters.

Building standard
extended project property
filters

```
QUERY_PRJ_VAL_StandardName = PropertyValue
QUERY_PRJ_VAL2_StandardName = PropertyValue
QUERY_PRJ_OP_StandardName = FilterOperator
```

StandardName is any one of the Standard Extended Project Property names, as described in the Introduction to the General Reference, with the PRJ_ prefix removed. These names are not case sensitive.

Here are some examples:

```
QUERY_PRJ_VAL_I_TASKID = 120
QUERY_PRJ_OP_I_TASKID = does not equal
```

Building extended
document property filters

```
QUERY_XDOC_VAL_ExtendedDocumentPropertyNameForCategory =
  PropertyValue
QUERY_XDOC_VAL2_ExtendedDocumentPropertyNameForCategory =
  PropertyValue
QUERY_XDOC_OP_ExtendedDocumentPropertyNameForCategory = Operator
```

ExtendedDocumentPropertyNameForCategory is the Bridge-encoded extended Document Property Name, as it appears in the Documanager Administrator for the intended Category. Note that *ExtendedDocumentPropertyNameForCategory* is case-sensitive, and is not the DB Field Name.

The Category corresponding to these Extended Document Properties must be supplied in the variable (DOC_CATEGORY). If *ExtendedDocumentPropertyNameForCategory* is not found for DOC_CATEGORY, it will be ignored.

Building basic document property filters

```
QUERY_DOC_VAL_PropertyName = PropertyValue
QUERY_DOC_VAL2_PropertyName = PropertyValue
QUERY_DOC_OP_PropertyName = FilterOperator
```

PropertyName is a document property name from the list given immediately below. This is the same as the property name in *dmg_api*, except *PropertyName* is not case sensitive.

Strings supported for *PropertyName* for basic document property filters, along with their property types:

PropertyName	Property Type
Name	String
Author	String
Category	String
Description	String
SubCategory	String
Status	String
Keyword1	String
Keyword2	String
Flag1	String
Flag2	String
Date	DateTime
FileType	String
AddedFrom	String
AddedOn	DateTime
DueDate	DateTime
CheckedOutBy	String
CheckedOutFor	String
Volume	String
MajorVersion	Long
MinorVersion	Long
LastEditedBy	String

LastEditedOn	DateTime
Id	ULong
Obsolete	Short
Released	Short
Approved	Short
ContentModifiedOn	DateTime
ContentSize	Long
LastAccessOn	DateTime
VersionComment	String
SourceDocumentID	ULong

Additional strings supported for *PropertyName* for basic document property filters when the bridge is connected to Documange Service Release 2 or higher:

PropertyName	Property Type
ContentsTag	Long
AnoAnnotationsTag	Long
RenditionID	Long
RenditionKey	String

Additional strings supported for *PropertyName* for basic document property filters when the bridge is connected to Documange Service Release 4 or higher and Dmg_Api 1.4.1 or higher is in use:

PropertyName	Property Type
RetentionDate	DateTime
ProtectionStatus	Short

Here is an example:

```
QUERY_DOC_VAL_Author = Landis
QUERY_DOC_OP_Author = does not equal
```

The above will cause a filter to be created that will find Documents whose Author Property is not Landis.

Building basic diary
property filters

```
QUERY_DIA_VAL_PropertyName = PropertyValue
QUERY_DIA_VAL2_PropertyName = PropertyValue
QUERY_DIA_OP_PropertyName = FilterOperator
```

PropertyName is a diary property name from the list given immediately below. This is the same as the property name in `dmg_api`, except *PropertyName* is not case sensitive.

Values allowed in *PropertyName* for basic diary property filters, along with their property types:

PropertyName	Property Type
Title	String
Description	String
Priority	Short
CreatedOn	DateTime
StartDate	DateTime
DueDate	DateTime
CompletedOn	DateTime
LastModifiedOn	DateTime
NextNotification	DateTime
AssignedBy	String
Owner	String
Attachment	String
Status	Short
DiaryID	String

Filter Operators

The values for *FilterOperator* are listed below by data type. They are case-insensitive.

For BETWEEN, two values must be supplied (`VAL_` and `VAL2_`) and the first value must be less than or equal to the second value. BETWEEN is inclusive.

For IN, the entire list of possible IN values must be specified. The list must be formatted correctly for its data type. String Properties must have the list items bracketed with a single quote (') (ASCII 39). List items must be separated with a comma. Extra spaces in the list are not significant. Note that when the value of a string Property contains a single quote character, such as `O'Brien`, when the value is put on the IN list, the single quote character must be escaped by entering it twice, as shown here:

```
'O''Brien'
```

The bridge does not validate the list of IN values. Since it is not validated, a subquery can be passed in for the value of the IN.

IS NULL and IS NOT NULL apply whether the INI option [NullValues] EnableNullValues is ON or OFF. If one of these operators is used, then SQL Builder ignores the value for the filter term and instead uses one of the special SQL predicates “IS NULL” and “IS NOT NULL”. Do not use these operators for fields that cannot be NULL, since a SQL error is likely to result. See [NULL Values as Inputs and Outputs on page 40](#) for more information.

Filter operators applicable to string properties:

```
EQUALS  
DOES NOT EQUAL  
BEGINS WITH  
CONTAINS  
IS LESS THAN  
IS GREATER THAN  
IN  
IS NULL  
IS NOT NULL
```

The IS LESS THAN and IS GREATER THAN operators for strings compare alphabetically in ascending order, for example, “Bob” is less than “Carol”.

Filter operators applicable to Date, Time, or DateTime properties:

```
EQUALS  
DOES NOT EQUAL  
BETWEEN  
ON OR BEFORE  
ON OR AFTER  
IS NULL  
IS NOT NULL
```

Note that, for a DateTime property, if only the date is given, the time defaults to 00:00:00. Because of this, ON OR BEFORE, say, “10/15/2002” for a DateTime property will be interpreted as ON OR BEFORE “10/15/2002 00:00:00”, which means that no items for 10/15/2002 will be found even if they exist.

Filter operators applicable to Numeric Properties:

```
EQUALS  
DOES NOT EQUAL  
BETWEEN  
IS LESS THAN  
IS GREATER THAN  
IN  
IS NULL  
IS NOT NULL
```

ORDERBY SYNTAX

The names of all variables for building orderby clauses begin with OB. You can specify as many clauses as desired. For each orderby clause, you must pass in a variable to specify the property to be ordered by. You may optionally pass in a variable to specify *OrderbyOperator*.

The recognized values for *OrderbyOperator* are ASC or DESC for “ascending” and “descending”. Additionally, any word or phrase beginning with ASC, case independent, is recognized as ASC and any word or phrase beginning with DESC, case independent, is recognized as DESC. *OrderbyOperator* is optional. The default if its value is not specified or not recognized is ASC.

In the syntax described below, the attachment variable name indicates the type of orderby clause being constructed. The *n* following OB indicates the order in which the clause is to be added to the filter. The order number always starts at 1. For example, OB1_DOC will cause a document orderby clause to be constructed and be the first clause in the orderby.

If OB*n* for a particular type of filter is not supplied or is empty, all higher numbers are ignored. For example, if OB1_DOC is supplied and OB2_DOC is not supplied or is empty, OB1_DOC will be used, but OB3_DOC and higher will be ignored even if they have been supplied.

The property names which are given as the values of the variables (*FolderPropertyName*, etc.) must not be bridge encoded. If an unrecognized property name is passed in, an error occurs. If duplicate property names are passed in (Example: OB1_DIA = Name, OB2_DIA = Name), an error occurs.

The cabinet is indicated by a separate variable (CABINET) in all the rules that will use this syntax.

Building folder order by

```
OBn_XFLD_Lx = FolderPropertyName
OBn_XFLD_Lx_OP = OrderbyOperator
```

In the above example, *x* is the one-based level number of the cabinet.

For XFLD orderby clauses, ordering is by Level, order numbers start at 1 within each level, and the ordering for Level 1, if any, is applied first, followed by the ordering for level 2, if any, and so on.

FolderPropertyName is the same as *FolderPropertyName* in QUERY_XFLD, except that it must not be Bridge encoded.

In this example, folders are primarily ordered by Last Name and secondarily by First Name for Level 1. For Level 2, Folders will be ordered by Due Date with the latest first. Notice that “Descending order” is a valid value for OB1_XFLD_L2_OP because it begins with the letters “Desc”.

```
OB1_XFLD_L1 = Last Name
OB2_XFLD_L1 = First Name
OB1_XFLD_L2 = Due Date
OB1_XFLD_L2_OP = Descending order
```

Building project order by

```
OBn_XPRJ = ProjectPropertyName
OBn_XPRJ_OP = OrderbyOperator
```

ProjectPropertyName is the same as *ProjectPropertyName* in QUERY_XPRJ, except that it must not be Bridge encoded. The use of this syntax is identical to the use of the syntax for building Folder orderby filters except for the Cabinet Level designator required when building Folder orderby filters.

Building document order by
by

```
OBn_DOC = PropertyName  
OBn_DOC_OP = OrderbyOperator
```

PropertyName is the same as *PropertyName* in QUERY_DOC. The same list of Properties applies.

Building diary order by

```
OBn_DIA = PropertyName  
OBn_DIA_OP = OrderbyOperator
```

PropertyName is the same as *PropertyName* in QUERY_DIA. The same list of Properties applies.

Here is an example:

In this example, Diaries are ordered by Priority. Since *OrderbyOperator* was not specified, they will be in ascending order.

```
OB1_DIA = Priority
```


DOWNLOADING FILES FROM THE CLIENT APPLICATION

The Documanager Bridge client application is responsible for writing a file being returned from a bridge transaction into a properly formed HTTP reply so that the web server can send it to the browser. This method of delivering the file to the browser provides good file security because the downloaded file is available only to a user who has gone through the bridge's user security mechanisms.

The names of all of the bridge rules that return files begin with "DmgBrsReturn", for example, DmgBrsReturnDocument. The Documanager Bridge client application must be prepared to download a file in response to any of these rules.

If your bridge application uses the standard CGI client, you can use the Documanager Bridge client rule DmgBrcDumpFile, which will write the file to an HTTP reply for you.

If you are using some other environment for your Documanager Bridge application, you can take advantage of the same attachment variables that DmgBrcDumpFile uses to obtain the necessary information about the file. These variables provide a path to the file (TARGET_FILE), the size of the file (HTTP_CONTENTLENGTH), and the file's MIME type (HTTP_MIMETYPE), all of which are needed when writing a file to an HTTP reply.

In addition to writing the file itself, your Documanager Bridge client application must also set the Bridge cookie in the HTTP header if [ValidateSession] UseCookies is ON. DmgBrcDumpFile does this automatically based on its inputs. Again, you can use these same inputs (the HTTP_COOKIE... variables) when setting the cookie. See [The Bridge Cookie on page 34](#), and DmgBrsValidateSession and DmgBrcDumpFile.

See [Serving Files from the Web Server on page 50](#) for an alternative way to download files to the browser.

SERVING FILES FROM THE WEB SERVER

In addition to the usual method of downloading files from the client application, Documanager Bridge has an alternative mechanism for downloading files to a browser, known as *bridge temp files*, that lets the web server serve files to a browser directly. It is called *temp files* because the files are served with temporary names and remain available for only a limited amount of time.

The advantage of using bridge temp files is that, for correctly configured files of type PDF and some other file types, certain web servers and browsers can cooperate to *byte serve* the files. That is, the server will stream such a file to the browser, and the browser will begin to display the file before it is completely downloaded. Especially in the case of large files, this boosts perceived performance.

Since read access must be granted on a directory containing files that the web server will serve directly, there are security considerations. It should not be possible for just any user to see these files, since the user may not have the requisite Documanager privileges to see any file. To address these security consideration, the files in the TempFiles Cache have unpredictable temporary names, and DmgBrsIdle deletes them periodically.

All of the rules whose names begin with *DmgBrsReturn* can output TempFiles. When any of these rules outputs a Temp File, it will change the ReqType to that configured in the INI option [ReqTypes] Redirect. See the section on Redirection.

For more information on configuring Bridge Temp Files, see the INI options for [TempCache], and the rule DmgBrcReceiveTempfile.

When the result of a bridge transaction is the serving of a bridge temp file, then the bridge cookie cannot be updated during that transaction because the web server has no knowledge of it.

TRANSFERRING FILES FROM THE BRIDGE SERVER TO THE CLIENT SERVER

When the IDS/Bridge server and IDS/Bridge client application are on the same machine, the issue of transferring files between them does not arise. However, most systems require that they be on separate machines either for security or for capacity reasons.

This section describes techniques for configuring your system for file transfer under these circumstances. Examples are for the standard CGI client, but the same principles apply to clients in other environments.

CHOOSING A FILE TRANSFER METHOD

There are two main methods for transferring files from the server to the client: shared directories, and the queue.

Shared directories

In this method, the server writes files to directories from which the client can read. You may use shared directories when there are no considerations preventing sharing, such as a firewall between the computers.

Queue

In this method, files are sent from the server to the client over the IDS queue. You may always choose to use the queue, and you must use the queue in situations where the client and server are on separate computers and shared directories are not feasible or not desired. The queue that you choose for your system must have the capacity and performance to handle file transfers in the volume that you expect.

CONFIGURING SHARED DIRECTORIES

The following topics explain how to transfer files from the server to the client using shared directories.

Configuring the TempCache Directory

If the INI option [TempCache] TempFiles is ON:

- 1 On the client, establish a directory to function as a temp files directory and share it. Do not use this directory or drive for other purposes.
- 2 In the web server, set up a virtual path for the temp files directory and allow read access.
- 3 On the Bridge server, set the option [TempCache] DirPath to the shared path of the temp files directory on the client. You may use the UNC path, or a path to a mounted network drive.
- 4 On the server, set the option [TempCache] DirURL to the virtual path of the temp files directory as usual.

Configuring the FileCache Directory

There are two approaches to configuring the cache directory.

- *Identical paths.* A shared FileCache directory is configured in such a way that it is accessible by the same path from the server and client computers.

- *Differing paths.* The shared FileCache directory is not configured to be accessible by the same path from the server and client computers. Instead, the client application is configured to construct the necessary path.

Using identical paths

To configure the FileCache directory with identical paths

- 1 On the client, establish a directory or drive to function as the FileCache directory and share it. Do not use this directory or drive for other purposes.
- 2 On the server, set the option [FileCache] DirPath to the shared path of the FileCache directory on the client. Make sure that it matches the path by which the directory is seen on the client. For example, if drive F is shared on the client, then mount it as drive F on the server, and set the option [FileCache] DirPath to F.

Using differing paths

To configure the FileCache directory with differing paths:

- 1 On the client, establish a directory to function as the FileCache directory and share it. Do not use this directory for other purposes.
- 2 On the server, set the option [FileCache] DirPath to the shared path of the FileCache directory on the client. You may use the UNC path, or a path to a mounted network drive.

NOTE: The attachment variable TARGET_FILE, produced by server rules that return files, is usually used by the client application to locate a cached file. However, TARGET_FILE contains the path to the file as seen by the server and thus will not work on the client if the path to the file on the client is different. On the client, the correct file path can always be derived by appending the value of the attachment variable TARGET_FILE_NAME.

Configuring a CGI client using differing paths

Set the optional parameters of the rule DmgBrcDumpFile in Docclnt.ini as follows:

- Set parameter 1 to the directory path of the shared FileCache, as seen from the client computer.
- Set parameter 2 to TARGET_FILE_NAME.
- Set parameter 3 to KEEP.

Here is an example: suppose the FileCache on the client computer is located at d:\docserve\cache. DmgBrcDumpFile would be configured as follows:

```
function = dmg_brw32-
>DmgBrcDumpFile,d:\docserv\cache\,TARGET_FILE_NAME,KEEP
```

TRANSFERRING FILES VIA THE QUEUE

Configuring the server

To configure the server for file transfers via the queue:

- 1 If the INI option [TempCache] TempFiles is ON, remove the DirPath option. Since the temp file is to be sent to the client, there is no need for a TempCache on the server.
- 2 Include the ATCSendFile rule on the rules lists in Docserv.ini that contain the rules that return files. Since the ATCSendFile rule sends the file on the reverse run, it may be placed near the top of the rules list so that it runs after most of the other rules have completed their forward and reverse runs.
- 3 For ATCSendFile, set the first parameter to a unique value such as DmgFile.
- 4 Set the second parameter to TARGET_FILE.
- 5 Set the third parameter to BINARY.

Here is an example:

```
[ ReqType:RETURN_DOC_AS_PDF ]
  function = atcw32->ATCLogTransaction
  function = atcw32->ATCLoadAttachment
  function = atcw32->ATCUnloadAttachment
  function = atcw32->ATCSendFile,DmgFile,TARGET_FILE,BINARY
```

Configuring a CGI client

To configure a CGI client for file transfer via the queue:

- 1 Include the ATCReceiveFile rule on rules lists in Docclnt.ini, just above DmgBrcDumpFile, for result types that return files.
- 2 Set ATCReceiveFile's first parameter to the same value as that for the corresponding ATCSendFile.
- 3 Set the second parameter to TARGET_FILE.
- 4 Set the third parameter to the path of the desired directory for temporarily storing target files, and use the "*.ext" format so that the ATCReceiveFile rule creates a temporary name. You can use any extension.
- 5 Do not set the fourth parameter, disposition. On the forward run, the ATCReceiveFile rule receives the file and the DmgBrcDumpFile rule sends the file to the web server. On the reverse run, the ATCReceiveFile rule deletes the file.

Here is an example:

```
[RESTYPE:RETURN_DOC]
  function = ATCw32->ATCLoadAttachment
  function = ATCw32->ATCApend2Attachment
  function = ATCw32->
    >ATCReceiveFile,DmgTarget,TARGET_FILE,e:\temp\*.tmp
  function = dmg_brw32->DmgBrcDumpFile
```

Configuring the redirection rules list

Additionally, if the option [TempCache] TempFiles is ON, you must configure the redirection rules list in Docclnt.ini to receive the file.

NOTE: The redirection rules list is the one whose ResType is configured in [ReqTypes] Redirect.

- 1 Put DmgBrcReceiveTempFile on the redirection rules list in Docclnt.ini.
- 2 Set its first parameter to match that of the corresponding ATCSendFile.
- 3 Set its second parameter to the directory on the client computer that the web server uses for serving temp files.
- 4 Optionally, put DmgBruPurgeFiles on the redirection rules list in Docclnt.ini to clean up expired temp files. If you have some other mechanism for cleaning up temp files, you do not need DmgBruPurgeFiles.
- 5 Set the first parameter of DmgBruPurgeFiles to match the directory in DmgBrcReceiveTempFile.
- 6 Set its second parameter to the number of minutes a file is allowed to remain in the temp directory before it is eligible for deletion.
- 7 Set its third parameter to TEMP_FILE_NAME. This causes DmgBruPurgeFiles to run only on those redirection transactions that involve temp files.

Here is an example:

```
[RESTYPE:BRI]
function = ATCw32->ATCLoadAttachment
function = ATCw32->ATCApPend2Attachment
function = pobrc-
>DmgBrcReceiveTempFile,DmgTemp,e:\docserv\html\temp\
function = dmg_brw32->DmgBruPurgeFiles, e:\docserv\html\temp\,8,
TEMP_FILE_NAME
```

For more information, see the example for DmgBrcReceiveTempFile.

CONVERTING DOCUMENTS

Documents of any type can be stored on Documanager. When retrieving the Document contents file of a checked out Document for the purpose of editing the file and updating the document with it, a user will obviously want the file in its original format. Use `DmgBrsReturnDocument` in this case, and in other situations in which the file is needed in its original format.

Sometimes, however, a user wants to retrieve a Document file simply for the purpose of looking at it in a browser. In this case, the Document may need to be converted into a format that a browser can display. For example, browsers don't know how to display Printstream files, but if correctly configured, most can display PDF files. So, Documanager Bridge provides a rule to convert Documents, `DmgBrsReturnConvertedDocument`.

This rule depends on libraries and rules that may be part of other bridges to perform conversions. See the description of `DmgBrsReturnConvertedDocument` for details on which libraries and rules are used in conversions, and which conversions are possible.

RENDERING PAGE IMAGES

The Bridge rule `DmgBrsReturnDocumentPage` returns images that represent a single page from a Document contents file. The images may be of type JPG, black and white TIF, image PDF, EMF, or DCZ (compressed EMF). The typical type is JPG. The others are used more rarely.

EMF and DCZ require a deprecated browser plug-in for display, and are included for backward compatibility. `DmgBrsReturnDocumentPage` can render pages in EMF and DCZ format only for content files of type DCD.

Other than the limitation for EMF and DCZ files, pages from any kind of Document content file that the Shuttle library can image can be rendered as any of the output types. Pages cannot be rendered for FPP or DPA files.

Rendering a page image as image PDF requires a configuration like that for rendering documents as image PDF. See “Converting Documents” and `DmgBrsReturnConvertedDocument`.

PROPERTY SETS AND PROPERTY SET LISTS

A number of Bridge Server rules allow the optional use of a Property Set List. The purpose of a Property Set List is to limit the amount of data that the rule will put on the Queue by limiting the number of Properties returned for items such as Documents or Folders to those actually used by the Bridge Application.

You may not need Property Sets. If you use a Queue that has a limited capacity or limited performance, then you should consider using Property Sets in your Bridge Application. If Queue capacity and performance are not considerations, then you have no need for Property Sets in your application and you can ignore them.

If you decide to use Property Sets, you may use the rule DmgBrsLogPropertySets to write all of the possible complete Property Sets for your Documanager system to the Bridge Log. You may then name the Property Sets as desired and remove whatever Properties you don't need. See DmgBrsLogPropertySets.

Property Set Lists and Property Sets are formatted similar to INI files. See the description of the INI option [PropertySets] PropertySetFile for information on configuring the location of your Property Sets.

The name of a Property Set or Property Set List must begin with an alphabetic character and can contain only alphabetic and numeric characters and the underscore. Note in particular that this means it cannot contain spaces, colons, or brackets. It may be up to 128 characters in length, case-sensitive.

PROPERTY SETS

Property Sets may be configured for Extended Folder Properties, Extended Project Properties, Basic Document Properties, and Extended Document Properties.

Property Sets and Property Set Lists are defined in INI file sections. Parts of these INI section names are delimited with colons.

The INI section name for a Property Set for Extended Folder Properties is:

```
[PS_XFLD_Lx:setname:cabname]
```

PS_XFLD_L = the keyword that identifies this type of Property Set

x = a number representing the one-based Level of the Cabinet

setname = the name of the Property Set. Must be unique within this keyword, level number, and Cabinet.

cabname = the Bridge-Encoded name of the Cabinet

This Property Set contains the Bridge-Encoded Extended Folder Property names, as they appear in the Documanager Administrator for the business data table that is mapped to this Level of the Cabinet. These are case sensitive.

The INI section name for a Property Set for Extended Project Properties is:

```
[PS_XPRJ:setname:cabname]
```

PS_XPRJ = the keyword that identifies this type of Property Set

setname = the name of the Property Set. Must be unique within this keyword.

cabname = the Bridge-Encoded name of the Workflow Cabinet

This Property Set contains the Bridge-Encoded Extended Project Property names, as they appear in the Documange Administrator for the business data table that is mapped to this cabinet. These are case sensitive. The Standard Extended Project Properties, as described in the introduction to the Rules Reference, are not included.

The INI section name for Basic Document Properties is:

```
[PS_DOC:setname]
```

PS_DOC = the keyword that identifies this type of Property Set

setname = the name of the Property Set. Must be unique within this keyword.

This Property Set contains the Bridge attachment variable names of Basic Document Properties. See the description of DmgBrsPresentDocument for a complete list of these variable names. These are not case sensitive.

The INI section name for Extended Document Properties is:

```
[PS_XDOC:setname:categoryname]
```

PS_XDOC = the keyword that identifies this type of Property Set

setname = the name of the Property Set. Must be unique within this keyword and Category.

categoryname = the Bridge-Encoded name of the Category

This Property Set contains the Bridge-Encoded Extended Document Property names, as they appear in the Documange Administrator for the Category. These are case sensitive.

Example 1:

A Property Set for Extended Document Properties. The Category is “Customer Accounts”. The name of the Property Set is “For_Cust_Accounts”. The only Extended Document Property on this list is “Account Number”. Notice the Bridge Encoding for the spaces in the name of the Category and of the Extended Document Property.

```
[PS_XDOC:For_Cust_Accounts:Customer$20Accounts]
Account$20Number
```

A different Property Set for the same Category. The Property Set name is “For_Overdue_Cust_Accounts”, which is distinct from the name “For_Cust_Accounts” for this Category, as it must be. There are three Extended Document Properties on this list.

```
[PS_XDOC:For_Overdue_Cust_Accounts:Customer$20Accounts]
Account$20Number
Overdue
In$20Collection
```

Example 2:

A Property Set for Extended Folder Properties. Notice that we can use the same name, “Authors”, for each Level since names must be unique only within PS_XFLD_Lx for a given Cabinet. In this case, we have given the Property Sets the same name as the Cabinet.

```
[PS_XFLD_L1:Authors:Authors]
First.Name
Last.Name

[PS_XFLD_L2:Authors:Authors]
Zip
```

Phone

Example 3:

A Property Set with four Basic Document Properties.

```
[PS_DOC:My_Basic_Doc_Propset]
  DOC_NAME
  DOC_AUTHOR
  DOC_CATEGORY
  DOC_FILETYPE
```

PROPERTY SET LISTS

Property Set Lists are referenced by rules such as DmgBrsQuery, and contain a list of Property Set names. Every Property Set List must have a name.

The INI section name for a Property Set List is:

```
[PS_LIST: listname]
```

PS_LIST = the keyword that identifies this section as a Property Set List

listname = the name of the Property Set List. Must be unique within Property Set List Names.

A Property Set List may contain the names of:

- up to one PS_XFLD Property Set for each Level of a Cabinet. Property Sets for only one Cabinet may be listed. If a PS_XPRJ Property Set is listed, no PS_XFLD Property Sets may be listed.
- up to one PS_XPRJ Property Set. If any PS_XFLD Property Sets are listed, a PS_XPRJ may not be listed.
- up to one PS_DOC Property Set.
- up to one PS_XDOC PropertySet for each Category name.

Note that there is no minimum contents for a Property Set List.

Each Property Set on a Property Set List is listed exactly as it appears between the brackets in the INI section that defines the Property Set. Do not include the brackets themselves.

Here are some examples:

Property Set names are taken from the examples for Property Sets. Both of these property sets can be defined because they have different names.

```
[PS_LIST:General]
  PS_XFLD_L1:Authors:Authors
  PS_XFLD_L2:Authors:Authors
  PS_XDOC:For_Cust_Accounts:Customer$20Accounts

[PS_LIST:Overdue]
  PS_XFLD_L1:Authors:Authors
  PS_XFLD_L2:Authors:Authors
  PS_XDOC:For_Overdue_Cust_Accounts:Customer$20Accounts
```

COMPOUND DOCUMENTS

A *compound document* is a Documanager document whose contents file contains other files within it. These other files are called *subdocuments*. There may be any number of subdocuments. In the bridge, they are referred to by number, starting at 1.

A *compound document plug-in* is a shared library that conforms to the Documanager Bridge compound document plug-in interface. For your bridge application to recognize a particular document type as a compound document, you must configure the control section [CdPlugins] in `dmg_brs.ini`, and you must use the plug-in corresponding to the compound document type.

The type of compound document that a plug-in can handle is part of the plug-in's file name. The file names of plug-ins are as follows:

- For Windows: `dmgcd_filetype.dll`
- For UNIX: `libdmgcd_filetype.so`

In these names, *filetype* is the extension of the file that the plug-in can handle. For example, for files of type “xyz”, the Windows plug-in would be called “`dmgcd_xyz.dll`”. All characters in the filenames must be lower case.

The Bridge Compound Document plug-in interface is public. If you want to write a plug-in for a custom compound document file type, contact Support for documentation on writing a plug-in. Alternatively, you may engage Skywire Software Services to write a plug-in for you.

The Documanager Bridge has rules specifically to handle Compound Documents, such as `DmgBrsPresentCompoundDocument`, `DmbBrsPresentSubdocument`, and `DmgBrsSetupCompoundDocument`. Several of the Bridge's other rules that handle Documents can handle both simple and Compound Documents, such as `DmgBrsReturnDocument` or `DmgBrsReturnConvertedDocument`. See the Bridge Rules Reference for more information about all the rules that handle simple and/or Compound Documents.

The bridge rules that return lists of Documents (`DmgBrsQuery`, `DmgBrsQueryDocuments`, `DmgBrsQueryDocumentRenditions`, and `DmgBrsPresentProject`) return an attachment output variable for each Document on the list that indicates whether the Document is a Compound Document. If you have configured your bridge application to recognize one or more file types as compound documents, you must pay attention to this variable. If the document is compound, you must direct any transaction based on that document to a rules list that contains rules that can handle compound documents. If it is not compound, you must direct such transactions to a rules list that can handle simple Documents.

NON-WEB APPLICATIONS

The most common use of Documanager Bridge is to web-enable access to Documanager, and the bridge provides many features intended to support that. However, there is no requirement that the bridge application be a web application. Documanager Bridge could, for example, be used to support an automated system that processed files and stored them on Documanager.

Configuring the Bridge for a non-web application isn't very different from configuring it for a web application, but there are a few INI options to consider. Set the INI option [BridgeURL] UnsecuredServerURL to the name or full path of your bridge application. For non-web applications, you will typically not use individual user security, so you can usually set the INI option [UserCredentials] Source to Bridge.

Cookies are probably not necessary, so set [ValidateSession] UseCookies to OFF. Because URL encoding is not an issue for non-web applications, you can set [Web] URLEncoding to OFF. [TempCache] TempFiles can almost certainly be OFF.

Chapter 4

Configuring the Documanager Bridge

Documanager Bridge has these configuration files:

- `dmg_brs.ini`—This configuration file contains options for the Bridge Server rules. It must be in the IDS directory in which the Bridge is installed.
- `dmg_brmime.ini`—This configuration file contains a list of MIME types used by certain Bridge Server rules. It must be in the IDS directory in which the Bridge is installed.
- `dmg_brc.ini`—This configuration file contains options for the Bridge Client rules. It must be in the directory in which the standard CGI client is installed.

Rules that are not part of Documanager Bridge but that Documanager Bridge uses to carry out file conversions may also use configuration files. See the documentation for those rules for specifics on those configuration files. The description of the rule `DmgBrsReturnConvertedDocument` lists these other rules.

Note that for all options in Bridge configuration files that are ON/OFF switches, you can enter ON, YES, ENABLED, or TRUE to indicate that the option is ON, and OFF, NO, DISABLED, or FALSE to indicate that the option is OFF, all case insensitive.

If you change `dmg_brs.ini` or `dmg_brmime.ini`, you must restart the Bridge if you want the change to take effect immediately, or you may wait for the rule `DmgBrsIdle` to detect the change and restart the Bridge for you.

THE DMG_BRG INI FILE

The following tables describe the control groups and options in the `dmg_brs.ini` file, which is used to configure the bridge server rules. These tables are meant to be read in conjunction with the documentation on the server rules that use the options they describe.

Keep in mind that since the `dmg_brs.ini` file may contain sensitive information (such as passwords), remove web server read privileges from its directory once you have set it up properly to make sure this information is not accessible from the web.

Any control group whose defaults are adequate for the needs of your bridge application may be omitted. For example, `[Web]` could typically be omitted. And some Bridge applications omit `[BridgeCredentials]` because they use *blank name* authentication for the Bridge account.

[Logging]

Use this control group to configure logging by the Documange Bridge Server rules.

Note that certain bridge server rules log certain things regardless of the setting of the `Log` option. For example, `DmgBrsInit` always logs its activity.

Option	Description
Log	Default OFF. Set this to ON or OFF to control whether logging occurs. Logging dramatically reduces the performance of a system, and should be used only as an aid in setup and to diagnose problems.
LogToFile	Default ON. Set this to ON to cause logged messages to be written to the file configured in <code>FilePath</code> .
LogToHTML	Default OFF. Set this to ON to cause an attachment record set called <code>_DCI_DEBUG</code> with the fields <code>TIME</code> and <code>INFO</code> to be output with each transaction. This record set may be used to display debugging information on an HTML page.
LogToMonitor	Default OFF. Set this to ON to cause logged messages to be written to the system debugger. If the IDS server has no debugger and the system debugger is not active, this option does nothing.
FilePath	Default <code>dmg_brs.log</code> . The file where logged messages appear if <code>LogToFile</code> is ON. May be a file name or a full file path.
FilterAddress	Default empty. Set this to an IP address to limit logging to those transactions for which the variable <code>REMOTE_ADDR</code> equals this IP address.
LogINIFilesAtStartup	Default ON. Set this to ON to cause <code>DmgBrsInit</code> to log the contents of the INI files listed in <code>[INIstolog]</code> . This provides a record of the settings in the files right in the Bridge log, for easy reference.
DeleteLogAtStartup	Default ON. Set this to ON to cause the file configured in <code>FilePath</code> to be emptied when the forward run of <code>DmgBrsInit</code> begins. Ignored when <code>RotateLogFiles</code> is ON.

RotateLogFiles	Default is ON. When ON, log files will be rotated. When OFF, a single log file will be written to, without limit, unless it is deleted at startup (because DeleteLogAtStartup is ON.) This option allows you to turn off rotation, perhaps because you want to use a tool such as logrotate or rotatelogs instead. Rotation means that the log file is written to until it reaches the size configured in MaxLogFileSizeInKiloBytes. At that point, it is renamed by appending "_1" to its name and a new log file is started. If there is already a file of that name with "_1" appended to its name, then the existing "_1" file is renamed "_2", etc., until the number of log files configured in MaxSavedLogFiles is exceeded. At that point, the oldest saved log file, the one with the highest number appended to the name, is discarded.
MaxLogFileSizeInKiloBytes	Default is 2000. Limits the size of the current rotated log file. The minimum is 100, so that if a value of less than 100 is configured, it will be ignored and 100 will be used. Ignored if RotateLogFiles is OFF.
MaxSavedLogFiles	Default is 5. Limits the number of saved log files, not including the current, unnumbered log file. The minimum is 0, which means there are no saved log files and the current log file is simply over-written when its size reaches MaxLogFileSizeInKiloBytes. Ignored if RotateLogFiles is OFF.

[INIsToLog]

Use this control group to list the INI files that DmgBrsInit will log during startup if [Logging] LogINIFilesAtStartUp is ON.

You may list INI files by name plus extension, or by full path. If no path is specified, DmgBrsInit will look for the file in the IDS directory.

Examples

```
[INIsToLog]
dmg_brs.ini
dmg_brmime.ini
dap.ini
c:\WINDOWS\poffice.ini
```

Configuring the Bridge Account

There are two alternative methods available for configuring Bridge credentials: with the [BridgeProfile] control group or with the [Documange] and [BridgeCredentials] control groups. Either of these alternatives may be used to configure the credentials that DmgBrsInit will use when logging in to Documange. These credentials are referred to as the "Bridge Account".

If IDS is running on Windows as an NT Service, that service cannot log on as *system account*. Instead, you must specify an account for the service by name and password.

If Documange is running on a computer other than the one running IDS, the account whose credentials are used must be a domain account visible to both computers.

If multiple Bridges are servicing the same Documanager system, each Bridge must have a different account.

Any account the Bridge uses to log into Documanager must have sufficient privileges to accommodate all users you want to have access. This is because the Bridge Account will be used for all access, even the access of individual users who log into the Bridge with the rule DmgBrsLogin. When a user logs in, DmgBrsLogin authenticates that user's credentials. On subsequent transactions involving that user, the Bridge Account impersonates the user, and the user's privileges can be no greater than those of the Bridge Account. For example, if the Bridge Account is a member of groups A and B, and a user is a member of groups A, B, and C, when that user logs into the Bridge, the user will have only those privileges associated with membership in groups A and B, and not those that are unique to C. This means that the Bridge Account must be a member of every group that you want to allow to access Documanager through the Bridge.

The fact no user of the Bridge can ever access anything that cannot be accessed through the Bridge Account provides a kind of security. If you do not want web users ever to be able to access things that only the Sales group can access, then simply make sure the Bridge Account is not a member of the Sales group.

Every group of which the Bridge Account is a member must have User Impersonation enabled, which is set up in the Documanager Administrator. To do this in the Administrator, click on the User Groups button. Next, select the desired group, and check 'Allow User Impersonation'. For more details, please see the Documanager Administrator's Guide.

[BridgeProfile]

Use this control group to configure the identity of the Documanager system and the credentials that the Bridge will use. Alternatively, you may use the [Documanager] and [BridgeCredentials] control groups and omit this control group.

Option	Description
Name	The name of a Documanager profile. Profiles can be created with the Dmgprofile Utility.

[Documanager]

When the [BridgeProfile] control group is not in use, use this control group to configure the identity of the Documanager system that the Bridge will use. All of these options are required and none have a default value. This control group is, along with [BridgeCredentials], an alternative to the [BridgeProfile] control group and is ignored if the Name option in [BridgeProfile] is set.

Option	Description
Router	The IP address or network identity of the Documanager Router.
Protocol	The protocol for communication with Documanager. Typically ncacn_ip_tcp.
Endpoint	The endpoint for communication with Documanager. Typically 4000.

[BridgeCredentials]

This control group is, along with [Documanager], an alternative to the [BridgeProfile] control group and is ignored if the Name option in [BridgeProfile] is set. Use it to specify the credentials that the Bridge will use to log into Documanager.

On Windows, you can omit the BridgeCredentials. In that case *blank name* authentication is used, which means that the credentials are automatically picked up from the account logged into the console if IDS is running as a console application, or from the account specified in the Windows Service Startup options for the IDS service if IDS is running as a service. On Windows, if UserName is empty, Password and Domain are ignored and blank name authentication is used.

Option	Description
UserName	Default empty. The user name for the Bridge Account.
Password	The password for UserName. This may be scrambled with the application dmg_brpwsram, available from Skywire Software Support.
Domain	The domain of UserName. Some systems may not require a domain.

[UserCredentials]

Use this control group to configure the way the rules DmgBrsLogin and DmgBrsValidateSession derive user credentials for each Bridge transaction.

Deciding how user credentials will be authenticated is an important part of configuring your Bridge system. You have these choices:

- If you want users to log into the Bridge with the Bridge rule DmgBrsLogin using their own credentials, set the Source option to Login. In this case, be sure to configure the options [ReqTypes] LoginUIDisplay, LoginExecute, and DestinationAfterLogin appropriately, and set up rules lists that correspond to those reqtypes.
- If you will authenticate a user's credentials by some means other than the Bridge rule DmgBrsLogin, such as the web server, then set the Source option to Trusted and configure the Trusted... options according to the way you will supply the user name and domain to DmgBrsValidateSession during Bridge transactions.
- If you want all users to use the Bridge account, then set the Source option to Bridge. In this case, all users will have the same privileges, no user name or domain needs to be supplied for individual users, and no user will need to log in.

Option	Description
Source	<p>Default is <i>Login</i>. The source of the user domain and name for each Bridge transaction. Recognized values, case independent, are:</p> <p><i>Login</i>—Derive user domain and name from those authenticated by the rule <i>DmgBrsLogin</i> and stored in the Bridge cookie.</p> <p><i>Trusted</i>—Derive user domain and name from the attachment variables specified in the “<i>Trusted...</i>” options below. These attachment variables must be present for each transaction, and must be on the attachment before <i>DmgBrsValidateSession</i> is called. Useful for “single sign-on” (user authentication by the web server) or for deriving valid credentials from sources other than Bridge rules.</p> <p><i>Bridge</i>—Use the domain and name of the Bridge account for all transactions, with no individual login.</p>
TrustedQualifiedNameVar	<p>The name of an attachment variable on the output attachment containing the qualified name (domain[delimiter]name) of the credentials to be trusted. Either this or <i>TrustedNameVar</i> and <i>TrustedDomainVar</i> (where applicable) must be specified if the <i>Source</i> option is <i>Trusted</i>. Ignored if the <i>Source</i> option is not <i>Trusted</i>, or if the <i>TrustedNameVar</i> option is not empty. Default is “<i>REMOTE_USER</i>”, which is the name of the typical Web environment variable that contains the validated user domain and name.</p>
TrustedQualifiedNameFormat	<p>The format of the value of the variable configured in the <i>TrustedQualifiedNameVar</i> option. The format is specified with two case independent keywords, “<i>domain</i>” and “<i>name</i>” separated by a single character delimiter that may be any character. The keywords may be in either order. There must not be any spaces or other characters between the keywords and the delimiter. Optional. The default is “<i>domain/name</i>”, which is a typical format used in IIS's <i>REMOTE_USER</i> variable on many windows platforms. Example: <i>name@domain</i>.</p>
TrustedNameVar	<p>The name of an attachment variable on the output attachment that contains the user name. Ignored if the <i>Source</i> option is not <i>Trusted</i>. Leave this option empty if you want <i>TrustedQualifiedNameVar</i> to be used. Optional. Default is empty.</p>
TrustedDomainVar	<p>The name of an attachment variable on the output attachment that contains the domain of the user. Ignored if the <i>Source</i> option is not <i>Trusted</i>. This is required only for those environments in which domains are used, and <i>TrustedNameVar</i> is used. Optional. Default is empty.</p>

Examples Here are some examples:

- Use the *BridgeCredentials* for all transactions.

```
[UserCredentials]
Source = Bridge
```

- To set up single sign-on when the web server is IIS, IIS is configured to require user login, and the standard CGI client is in use (which puts webserver environment variables into attachment variables of the same name as the environment variables), the following settings are sufficient because of the defaults.

```
[UserCredentials]
Source = Trusted
```

- In this example, assume that something outside of the Bridge will put the authenticated user name and domain into a variable called VALID_USER in the format domain.name, that is, delimited by a period.

```
[UserCredentials]
Source = Trusted
TrustedQualifiedNameVar = VALID_USER
TrustedQualifiedNameFormat = Domain.Name
```

- In this example, assume that something outside of the Bridge will put the authenticated user name into a variable called VALID_USER, and that domains are not in use in this environment.

```
[UserCredentials]
Source = Trusted
TrustedNameVar = VALID_USER
```

[BridgeURL]

Use this control group to configure the way DmgBrsLogin and DmgBrsValidate session build the URL to Documanager Bridge.

Option	Description
ProtocolSpecifier	The default is HTTP. The web protocol. May also be HTTPS, or another protocol.
SecureServerUsage	Default is OFF. Recognized values are ON, Login, or OFF. Controls whether Bridge rules use SecuredServerURL to build the Bridge URL. When SecureServerUsage is ON, SecuredServerURL is always used. When SecureServerUsage is Login, it is used only for the transactions whose reqtypes are [ReqTypes] LoginUIDisplay and LoginExecute. When it is OFF, it is never used.

SecuredServerURL	Required when SecureServerUsage is ON or Login, otherwise it is ignored. The fully qualified URL of the secured server, including the port. For example: https://123.45.45.450:143/dcltw32.exe Instead of a URL, you can enter the name of any client application to be used with the Bridge, useful in non-web applications.
UnsecuredServerURL	Required when SecureServerUsage option is Login, ignored when SecureServerUsage is set to On, and optional when SecureServerUsage is set to Off. The fully qualified URL of the unsecured server, including the port if it is not 80. This URL must be in the same domain as that of the SecuredServerURL when the SecureServerUsage option is set to Login. Include the port if the port is not 80. For example: http://123.45.45.450/dcltw32.exe Instead of a URL, you can enter the name or full path of any client application to be used with the Bridge, useful in non-web applications.
SuppressPortInServerURL	Default is OFF. Set this to ON to cause the Bridge rules to omit the port when they construct a URL. This can be useful when a load balancing switch directs traffic to various web servers and the load balancing switch requires that the port not be included in the URL.

[ValidateSession]

Use this control group to configure options for the rule DmgBrsValidateSession.

Option	Description
UseCookies	Default is ON. Set this to OFF to keep the rule from reading and writing cookie data. This must be set to ON if [UserCredentials] Source is Login, or the Bridge will not initialize.
ScrambleCookies	Default is ON. This may be set to OFF temporarily for testing purposes, but should be ON for normal operation.
CookieEncryptionKey	Default is empty. Any text in this option will be used to scramble the cookie text. If no CookieEncryptionKey text is provided, a default key is used. Up to 256 characters are used, and that is the recommended size.
CookiesPersist	Default is ON. The default value for HTTP_COOKIEPERSISTS.
CookieDateFormat	Default is HTTP. The default value for COOKIE_DATE_FORMAT. May be either HTTP or ASP.

UserSessionTimeoutInMinutes	Default is 0. Set this to the maximum length of time allowed between Bridge transactions for a user when [UserCredentials] Source is Login. A value of 0 means “never time out”. If this time is exceeded, the rule will initiate an Ad Hoc Login.
LogoutUsersOnBridgeRestart	Default is ON. When [UserCredentials] Source is Login, this determines whether a user must log in again after the Bridge has been restarted even if the user’s cookie is valid otherwise.
ShowGroupAccess	Default is OFF. Controls whether this rule outputs the attachment variable SESS_GROUPACCESS. There is a performance cost to determining a user’s group access. Set this to ON only if your Bridge application requires it. See the description of DmgBrsValidateSession in the Rules Reference for more information.

[FileCache]

Use this control group to configure options that deal with the Bridge File Cache. These options affect DmgBrsInit and DmgBrsIdle, and any rule that deals with files. The Bridge File Cache is a directory in which the Bridge temporarily stores Document contents files, Annotations files, and files derived from those files, such as images of individual Document pages or converted Documents. This directory is vital to the operation and good performance of the Bridge, and must not be used for any other purpose.

Option	Description
DirPath	Required. The full path of the directory to use as the Bridge File Cache. This must not be the IDS directory.
PurgeAtStartUp	Default is ON. Set this to ON to cause DmgBrsInit to delete all files in DirPath when the Bridge starts up.
MaxSizeInMegabytes	Default is 1000 (roughly a gigabyte). Maximum is 4095. If the Bridge File Cache is larger than the maximum, DmgBrsIdle will delete files, least recently used first, until it is reduced to the maximum, but will never delete files younger than MinPurgeAgeInMinutes.
MinPurgeAgeInMinutes	Default is 10. DmgBrsIdle never deletes files younger than this from the Bridge File Cache.
MaxPurgeAgeInMinutes	Default is 525600 (one year). DmgBrsIdle always deletes files this old or older from the Bridge File Cache. MaxPurgeAgeInMinutes must be greater than MinPurgeAgeInMinutes.

Various rules store files in the Bridge File Cache. The names of the files in the Bridge File Cache are not part of the public interface of the Bridge and may change at any time. If you need to access Document contents files in the cache, use the attachment variables whose names begin with “DOCFILE_”, which are output by the rules that store Document contents files in the Cache.

[TempCache]

Use this control group to configure options that deal with the Bridge Temp File Cache. These options affect DmgBrsIdle, and all rules whose names begin with “DmgBrsReturn”.

Option	Description
TempFiles	Default is OFF. Set this to ON to cause all rules whose names begin with “DmgBrsReturn” to output Temp Files, and DmgBrsIdle to purge the directory DirPath. TempFiles ON takes effect only if DirPath is configured.
DirPath	Default is empty. The full path of the directory to use as the Bridge Temp File Cache. This must not be the IDS directory, and must not be the Bridge File Cache.
DirURL	Required if TempFiles is ON. The complete URL of DirPath, without a trailing delimiter. The web server must be configured to allow read access from this directory, but it should not allow browsing of this directory. Example: http://10.7.10.91/bridgetempfiles
PurgeAgeInMinutes	Default is 10. DmgBrsIdle purges files this old or older from the Bridge Temp Files Cache. Set this time to be long enough to assure that a Temp File has enough time to be downloaded to the browser before it is deleted.

[CdPlugins]

Use this control group to configure the way bridge rules find Compound Document Plug-ins.

Option	Description
DirPath	The path of the directory that contains Compound Document Plug-ins. This must not be the IDS directory or the Bridge FileCache or TempCache directories. However, it may be a subdirectory of the IDS directory. Leave this option empty if you do not use Compound Document Plug-ins. If you use Compound Document Plug-ins, configure this directory and put your Plug-ins into the directory. Optional.

[ReqTypes]

Use this control group to configure certain IDS reqtypes that Bridge rules need to know about.

In Documanager Bridge, you are free to use any ReqType you like. However, there are a few cases where the Bridge needs to know about the ReqTypes you have chosen so that it can detect what kind of transaction is occurring, or so that it can change the ReqType appropriately to signal certain conditions. Use these options to configure those ReqTypes.

Option	Description
Failure	Default is BXX. Any bridge server rule that signals failure will do so by changing the reqtype to this. See “How Server Rules Signal An Error”.
Redirect	Default is BRI. Whenever a Bridge server rule must control what the subsequent transaction does, it will do so by changing the ReqType to this. The Bridge application must respond by redirecting to the URL in the variable BRS_DESTINATION. This is used only if [UserCredentials] Source is Login, or if [TempCache] TempFiles is ON. See “Redirection”.
LoginUIDisplay	Default is BLD. The reqtype that is used to present the user with a web page soliciting their credentials. This is used only if [UserCredentials] Source is Login.
LoginExecute	Default is BLE. The reqtype that is used to run the rule DmgBrsLogin. This is used only if [UserCredentials] Source is Login.
DestinationAfterLogin	Default is BWD. The reqtype to which the rule DmgBrsLogin will redirect the user after login if it was not an Ad Hoc Login. Use this to control what happens after a user logs in with DmgBrsLogin. This is used only if [UserCredentials] Source is Login.

[PropertySets]

Use this control group to configure the location of Property Sets used in DmgBrsDeleteDocument, DmgBrsGetTaskQueue, DmgBrsPresentDocument, DmgBrsPresentProject, DmgBrsQuery, DmgBrsQueryDocuments.

Option	Description
PropertySetFile	Default is dmg_brs.ini. The file in which Property Sets are configured. Notice that rules that use Property Set Lists will look in dmg_brs.ini for them if this option is not configured.

[NullValues]

Use this control group to configure the way that the Bridge handles NULL database values for Extended Folder, Project, and Document Properties. See the section “NULL Values as Inputs and Outputs”.

Option	Description
EnableNullValues	Default is OFF. To cause the Bridge to use and recognize the string in the option NullValue, this must be set to ON. Otherwise, the Bridge will output a NULL numeric value as 0, and a NULL string value as an empty string, and will not recognize the string set in NullValue as a NULL value on input.
NullValue	Default is [NULL], including the brackets. This is the string that represents NULL values. You may set it to any string, although something short is recommended. Choose this string carefully so as not to coincide with any actual data value the field might possibly have. This option is ignored if EnableNullValues is OFF.

[Web]

There are a few output attachment variables, such as QUERY_STEM in DmgBrsQuery, that Bridge server rules URL encode by default. They are noted in the documentation where they're described. Use this control group if you want to turn off automatic encoding. If your Bridge Application is a non-web application, you will probably want to set URLEncoding to OFF.

Option	Description
URLEncoding	Default is ON. May be turned off if desired for a non-web application.

[Query]

Use this control group to configure options for DmgBrsQuery.

Option	Description
MaxHits	Default is 20. The default and limit for MAXHITS.
FldsOnly	Default is OFF. The default for FLDS_ONLY.
QueryVersions	Default is OFF. The default for QUERY_VERSIONS.
AllowFldFilter	Default is OFF. Must be ON to use FLD_FILTER, or the rule will return an error if FLD_FILTER exists.
AllowDocFilter	Default is OFF. Must be ON to use DOC_FILTER, or the rule will return an error if DOC_FILTER exists.
AllowVerFilter	Default is OFF. Must be ON to use VER_FILTER, or the rule will return an error if VER_FILTER exists.
AllowXDocFilter	Default is OFF. Must be ON to use XDOC_FILTER, or the rule will return an error if XDOC_FILTER exists.

AllowFldOrderBy	Default is OFF. Must be ON to use FLD_ORDERBY, or the rule will return an error if FLD_ORDERBY exists.
AllowDocOrderBy	Default is OFF. Must be ON to use DOC_ORDERBY, or the rule will return an error if DOC_ORDERBY exists.
AllowFullTextFilter	Default is OFF. Must be ON to use FULLTEXT_FILTER, or the rule will return an error if FULLTEXT_FILTER exists.

[QueryDocuments]

Use this control group to configure options for DmgBrsQueryDocuments.

Option	Description
MaxHits	Default is 20. The default and limit for MAXHITS.
GroupHitsByLevel	Default is OFF. The default for GROUP_HITS_BY_LEVEL.
FldsOnly	Default is OFF. The default for FLDS_ONLY.
QueryVersions	Default is OFF. The default for QUERY_VERSIONS.
AllowFldFilter	Default is OFF. Must be ON to use FLD_FILTER, or the rule will return an error if FLD_FILTER exists.
AllowDocFilter	Default is OFF. Must be ON to use DOC_FILTER, or the rule will return an error if DOC_FILTER exists.
AllowVerFilter	Default is OFF. Must be ON to use VER_FILTER, or the rule will return an error if VER_FILTER exists.
AllowXDocFilter	Default is OFF. Must be ON to use XDOC_FILTER, or the rule will return an error if XDOC_FILTER exists.
AllowFldOrderBy	Default is OFF. Must be ON to use FLD_ORDERBY, or the rule will return an error if FLD_ORDERBY exists.
AllowDocOrderBy	Default is OFF. Must be ON to use DOC_ORDERBY, or the rule will return an error if DOC_ORDERBY exists.
AllowFullTextFilter	Default is OFF. Must be ON to use FULLTEXT_FILTER, or the rule will return an error if FULLTEXT_FILTER exists.

[QueryProjects]

Use this control group to configure options for DmgBrsQueryProjects.

Option	Description
MaxHits	Default is 20. The default and cap for MAXHITS.

CheckedOutState	Default is ANY. The default for CHECKEDOUT_STATE. Recognized values are ANY, CHECKED_OUT, and NOT_CHECKED_OUT, case independent. An unrecognized value is taken to mean ANY.
ProjectState	Default is ANY. The default for PROJECT_STATE. Recognized values are ANY, PENDING, and SUSPENDED, case independent. An unrecognized value is taken to mean ANY.
AllowPrjFilter	Default is OFF. Must be ON to use PRJ_FILTER, or the rule will return an error if PRJ_FILTER exists.
AllowPrjOrderBy	Default is OFF. Must be ON to use PRJ_ORDERBY, or the rule will return an error if PRJ_ORDERBY exists.

[QueryAllProjects]

Use this control group to configure options for DmgBrsQueryAllProjects.

Option	Description
MaxHits	Default is 20. The default and cap for MAXHITS.
CheckedOutState	Default is ANY. The default for CHECKEDOUT_STATE. Recognized values are ANY, CHECKED_OUT, and NOT_CHECKED_OUT, case independent. An unrecognized value is taken to mean ANY.
ProjectState	Default is ANY. The default for PROJECT_STATE. Recognized values are ANY, PENDING, and SUSPENDED, case independent. An unrecognized value is taken to mean ANY.

[QueryDocumentRenditions]

Use this control group to configure options for DmgBrsQueryDocumentRenditions.

Option	Description
ObsoleteFilter	Default is "NONE". The default for OBSOLETE_FILTER. Recognized values are ON, OFF, or "NONE", case independent, where ON and OFF may be indicated with the usual range of values for ON and OFF for INI options. An unrecognized value is taken to mean "NONE".
ReleasedFilter	Default is "NONE". The default for RELEASED_FILTER. Recognized values are ON, OFF, or "NONE", case independent, where ON and OFF may be indicated with the usual range of values for ON and OFF for INI options. An unrecognized value is taken to mean "NONE".

ApprovedFilter	Default is "NONE". The default for APPROVED_FILTER. Recognized values are ON, OFF, or "NONE", case independent, where ON and OFF may be indicated with the usual range of values for ON and OFF for INI options. An unrecognized value is taken to mean "NONE".
LatestVersionsOnly	Default is OFF. The default for LATEST_VERSIONS_ONLY.

[ListUsers]

Use this control group to configure options for DmgBrsListUsers.

Option	Description
MaxUsers	Default is 40. The default and cap for MAX_USERS.

[PresentDocument]

Use this control group to configure options for DmgBrsPresentDocument.

Option	Description
ShowVersions	Default is OFF. The default for SHOW_VERSIONS.
MaxVersions	Default is 10. The default and limit for MAX_VERSIONS.
ShowPages	Default is OFF. The default for SHOW_PAGES.
ShowActions	Default is OFF. The default for SHOW_ACTIONS.

[PresentCompoundDocument]

Use this control group to configure options for DmgBrsPresentCompoundDocument.

Option	Description
ShowVersions	Default is OFF. The default for SHOW_VERSIONS.
MaxVersions	Default is 10. The default and limit for MAX_VERSIONS.
ShowActions	Default is OFF. The default for SHOW_ACTIONS.
MaxSubdocuments	Default is 20. The default for MAX_SUBDOCS.
ShowSubdocsPages	Default is OFF. The default for SHOW_SUBDOCS_PAGES.

[PresentSubdocument]

Use this control group to configure options for DmgBrsPresentSubdocument.

Option	Description
ShowSubdocsPages	Default is OFF. The default for SHOW_SUBDOCS_PAGES.

[ReturnDocument]

Use this control group to configure options for DmgBrsReturnDocument.

Option	Description
Decompress	Default is OFF. The default for DECOMPRESS. Windows only.

[DocsIndicator]

Use this control group to configure SHOW_DOCS_IND for DmgBrsQuery and DmgBrsGetTaskQueue.

Option	Description
ShowDocsIndicator	Default is OFF. The default for SHOW_DOCS_IND.

[CreateDocument]

Use this control group to configure options for DmgBrsCreateDocument and DmgBrsUpdateDocumentFile.

Option	Description
DeleteFile	Default is ON. The default for DOC_DELFILE.

[UpdateAnnotations]

Use this control group to configure options for DmgBrsUpdateAnnotations.

Option	Description
DeleteFile	Default is ON. The default for ANNOTS_DELFILE.

[CheckOutDocument]

Use this control group to configure options for DmgBrsCheckOutDocument.

Option	Description
UsePersonalCabinet	Default is OFF. The default behavior when a COPY_CABINET is not specified. If OFF, it means “check out in place”. If ON, it means “check out to personal cabinet”. The default value for USE_PERSONAL_CABINET.

[CheckInDocument]

Use this control group to configure options for DmgBrsCheckInDocument.

Option	Description
VersionComment	Default is empty. The default for DOC_VERSIONCOMMENT.
CheckInType	Default is MAJOR. Recognized values are MAJOR, MINOR, and SAME, case independent. The default for CHECKIN_TYPE.

[PresentFolder]

Use this control group to configure options for DmgBrsPresentFolder.

Option	Description
ShowNumDocs	Default is OFF. The default for SHOW_NUM_DOCS.
ShowAncestors	Default is OFF. The default for SHOW_ANCESTORS.
ShowActions	Default is OFF. The default for SHOW_ACTIONS.

[SetupFolder]

Use this control group to configure options for DmgBrsSetupFolder.

Option	Description
UniqueFolder	Default is ON. The default for UNIQUE_FOLDER.
AllowFldFilter	Default is OFF. Must be ON to use FLD_FILTER, or the rule will return an error if FLD_FILTER exists.

[DeleteFolder]

Use this control group to configure options for DmgBrsDeleteFolder.

Option	Description
DeleteChildren	Default is OFF. The default for DELETE_CHILDREN.
DeleteDocuments	Default is OFF. The default for DELETE_DOCUMENTS.

[PresentProject]

Use this control group to configure options for DmgBrsPresentProject.

Option	Description
MaxDocumentHits	Default is 20. The default and limit for MAXDOCS.
ShowDocs	Default is OFF. The default for SHOW_DOCS.
ShowTeam	Default is OFF. The default for SHOW_TEAM.
ShowActions	Default is OFF. The default for SHOW_ACTIONS.
ShowHistory	Default is OFF. The default for SHOW_HISTORY.

[GetTaskQueue]

Use this control group to configure options for DmgBrsGetTaskQueue.

Option	Description
MaxProjects	Default is 20. The default and limit for MAXPROJECTS.
AutoFilter	Default is empty. The default for AUTO_FILTER. Can be configured to empty, for no filter, or CHECKED_OUT, case insensitive.

[Notes]

Use this control group to configure options affecting Notes for DmgBrsPresentFolder, DmgBrsPresentProject, DmgBrsQuery and DmgBrsGetTaskQueue.

Option	Description
MaxNotes	Default is 10. The default and limit for MAXNOTES in DmgBrsPresentFolder and DmgBrsPresentProject.
ShowNotes	Default is OFF. The default for SHOW_NOTES in DmgBrsPresentFolder and DmgBrsPresentProject.
ShowNotesIndicator	Default is OFF. The default for SHOW_NOTES_IND in DmgBrsQuery.
NotesAutoFilter	Default is ALL. The default for the NOTES_AUTO_FILTER input attachment variable in DmgBrsPresentFolder and DmgBrsPresentProject. Recognized values are ALL, CURRENT, and DELETED, case independent. Any unrecognized value is interpreted as ALL.

[CreateDiary]

Use this control group to configure options for DmgBrsCreateDiary.

Option	Description
MakeFilter	Default is OFF. The default for MAKE_FILTER.

[QueryDiaries]

Use this control group to configure options for DmgBrsQueryDiaries.

Option	Description
MaxDiaries	Default is 20. The default and limit for MAX_DIARIES.
AllowDiaryFilter	Default is OFF. Must be ON to use DIARY_FILTER, or the rule will return an error if DIARY_FILTER exists.
AllowDiaryOrderBy	Default is OFF. Must be ON to use DIARY_ORDERBY, or the rule will return an error if DIARY_ORDERBY exists.

[Rendering]

Use this control group to configure options that affect how page images or converted Documents are rendered by various rules, as noted.

Option	Description
PageConversion	Default is JPG. The default FileType for the conversion of an individual page of a Document. Used by DmgBrsPresentDocument, DmgBrsPresentDocumentPage, and DmgBrsReturnDocumentPage. Valid values are JPG, TIF, PDF, EMF, and DCZ.
DocumentConversion	Default is PDF. The default FileType for the conversion of a Document. Used by DmgBrsQuery, DmgBrsQueryDocuments, DmgBrsPresentDocument and DmgBrsReturnConvertedDocument.
DCDEnhancedPageDisplay	Default is OFF. Used by DmgBrsPresentDocument and DmgBrsPresentDocumentPage to determine if a page from a DCD file should be returned in the enhanced format EMF. This over-rides PageConversion for DCD pages.
CompressEnhancedDCD	Default is OFF. Used by DmgBrsPresentDocument and DmgBrsPresentDocumentPage when DCDEnhancedPageDisplay to determine if the EMF should be compressed to DCZ.
INSODirPath	Default is empty. The path to the directory containing the OutsideIn library, used by the Shuttle library to derive images from many types of files. During Bridge startup, DmgBrsInit looks for the library in this directory. If not found there, it looks in a sub-directory called "inso" in the IDS directory. If not found there, it does not cause Shuttle to initialize the OutsideIn library. Windows only.
DPAExtension	Default is DPA. Use this to configure the extension that identifies a file as DPA if it is different from DPA on your system.

[ReturnDocumentPage]

Use this control group to configure options for DmgBrsReturnDocumentPage.

Option	Description
MissingFile	Default is missing.jpg. The name or full path to a JPG file to use if a Document page cannot be created.
MaxWidthInPixels	Default is 0. The maximum width in pixels for page images in raster format. There is no maximum for EMF images. 0 means that no maximum should be applied.
MaxHeightInPixels	Default is 0. The maximum height in pixels for page images in raster format. There is no maximum for EMF images. 0 means that no maximum should be applied.
UseTPDForPDF	Default is OFF. Set this option to ON to cause the rule to set up the TPD rules for making image PDF files. Otherwise, the rule will use its built-in facility to make image PDF files.

[ReturnConvertedDocument]

Use this control group to configure options for DmgBrsReturnConvertedDocument.

Option	Description
PrintstreamToImagePDF	Default is OFF. The default for PRINSTREAM_TO_IMAGE_PDF.
PrintstreamToImagePDFColor	Default is OFF. Set this option to ON to allow print stream documents to be converted to image PDF files in color when they contain color. OFF forces the output PDF to be in black and white even if the document contains color. See the caution in the description of DmgBrsReturnConvertedDocument before setting this option to ON.
UseTPDForImagePDF	Default is OFF. Set this option to ON to cause the rule to set up the TPD rules for making image PDF files. Otherwise, the rule will use its built-in facility to make image PDF files.
PrintstreamConfig	Default is empty. The default for PRINTSTREAM_CONFIG.
CopyConfigFile	Default is isicpy.ini. The default for the value of the output attachment variable COPYCONFIGFILE when COPYCONFIGFILE doesn't already exist and when there is no COPYCONFIGFILE_x for the extension of the converted file.

Option	Description
CopyConfigFile_x	The value for the output attachment variable COPYCONFIGFILE when COPYCONFIGFILE doesn't already exist, where x is a file extension of the converted file, without a dot. There may be any number of these, but they must be unique. For example, the value of an option called COPYCONFIGFILE_XML would be used if the file extension of the converted file was XML.
MissingBlackAndWhitePageFile	Default is missingpdf.tif. Used when converting to image PDF and a TIF page cannot be created. Must be a single page black and white TIF file.
MissingColorPageFile	Default is missingpdf.jpg. Used when converting to image PDF and a JPG page cannot be created.]
CacheFileSuffixes	Optional. Default is empty. A list of allowable values for the input attachment variable CACHE_FILE_SUFFIX. The values must be separated by a comma, and can contain only lowercase alphanumeric characters or the underscore character. Bridge startup will fail if it detects any invalid characters. See the description of CACHE_FILE_SUFFIX under DmgBrsReturnConvertedDocument in the Bridge Rules Reference. Although there is no explicit limit on the number of suffixes in the list or the length of a suffix, we strongly recommend keeping the length of suffixes short to avoid the risk of a file path of illegal length. Example: CacheFileSuffixes = dml, dml2, ab_a, ab_c,_wm.

[ThumbNails]

Use this control group to configure thumbnailing options for DmgBrsPresentDocument and DmgBrsReturnThumbnail.

Option	Description
Thumbnails	Default is OFF. Set this to ON to cause DmgBrsPresentDocument to generate fields in the PAGES record set that describe thumbnail images of the pages, and to allow DmgBrsReturnThumbnail to create a thumbnail image.
MaxThumbnails	Default is 0. The maximum number of thumbnail images of pages that DmgBrsPresentDocument should describe in the PAGES record.
MissingFile	Default is missingthumb.jpg. The name or path to a JPG file to use if a thumbnail image for a particular page can't be made or isn't supposed to be made.

MaxWidthInPixels	Default is 64. The maximum width for a thumbnail image generated by DmgBrsReturnThumbnail.
MaxHeightInPixels	Default is 64. The maximum height for a thumbnail image generated by DmgBrsReturnThumbnail.
Darkness	Default is 40. The darkness for a thumbnail image generated by DmgBrsReturnThumbnail. Higher numbers are darker.

[DmgDiagnostics]

Use this control group to configure options for Documanager Diagnostic Tests. These tests may be run by the rules DmgBrsInit, DmgBrsIdle, DmgBrsRunDocumanagerDiagnostic, and by any server rule that fails. Running Documanager Diagnostic Tests is optional. Simply omit this control group if you do not want to run tests. See the Documanager documentation for information on Documanager Diagnostic Tests.

Option	Description
StartupDiagnostic	Default is empty. The name of a Documanager Diagnostic Test to be run by DmgBrsInit, or by the rule DmgBrsCheckPausedServer. If empty, no test is run.
IdleDiagnostic	Default is empty. The name of a Documanager Diagnostic Test to be run by DmgBrsIdle. If empty, no test is run.
ErrorDiagnostic	Default is empty. The name of a Documanager Diagnostic Test to be run when there is an error in a Bridge rule, since many errors may be due to a problem with Documanager. If empty, no test is run.
TestFailDiagnostic	Default is empty. A dummy test name. For the use of this option, see DmgBrsRunDocumanagerDiagnostic.
OnFail_Startup	Default is empty. The IDS action to take if StartupDiagnostic fails. Valid values are PAUSE, STOP, RESTART, case insensitive.
OnFail_Idle	Default is empty. The IDS action to take if IdleDiagnostic fails. Valid values are the same as for OnFail_Startup.
OnFail_ErrorDiagnostic	Default is empty. The IDS action to take if ErrorDiagnostic fails. Valid values are the same as for OnFail_Startup.

THE DMG_BRMIME INI FILE

The `dmg_brmime.ini` file contains a list of extensions and corresponding MIME types. The use of this INI file is optional. If you use it, place it in the same directory as the `dmg_brs.ini` file.

If this INI file is present, a Documanager Bridge Server rule looks in the INI file to determine the MIME type of a file that it is returning through the variable `TARGET_FILE`. If it does not find the MIME type, the Bridge uses its built-in list of types. An example `dmg_brmime.ini` file is installed with the Bridge which includes an expanded list of types. You can add, edit, or delete the types in this file as necessary. The format of an entry is:

```
[extension] = [MIMEtype]
```

Here is an example:

```
txt = text/plain
```

An extension can appear at most once in the list.

Generally, you will want to have this file match the MIME types at the web server, although this is not required.

THE DMG_BRC INI FILE

The following table describes the control group and options in the `dmg_brc.ini` file, which is used to configure Documanager Bridge Client rules.

[Logging]

Use this control group to configure logging by the Documanager Bridge Client rules.

Option	Description
Log	Default OFF. Set this to ON or OFF to control whether logging occurs. Logging dramatically reduces the performance of a system, and should be used only as an aid in setup and to diagnose problems.
LogToFile	Default ON. Set this to ON to cause logged messages to be written to the file configured in <code>FilePath</code> .
LogToHTML	Default OFF. Set this to ON to cause an attachment record set called <code>_DCL_DEBUG</code> with the fields <code>TIME</code> and <code>INFO</code> to be output with each transaction. This record set may be used to display debugging information on an HTML page.
LogToMonitor	Default OFF. Set this to ON to cause logged messages to be written to the system debugger. If the IDS server has no debugger and the system debugger is not active, this option does nothing.
FilePath	Default <code>dmg_brc.log</code> . The file where logged messages appear if <code>LogToFile</code> is ON. May be a file name or a full file path.
FilterAddress	Default empty. Set this to an IP address to limit logging to those transactions for which the variable <code>REMOTE_ADDR</code> equals this IP address.

Chapter 5

Troubleshooting and Diagnostics

There are several tools for troubleshooting performance and functional problems with Documanage Bridge.

These tools include:

- [Bridge Logging on page 88](#)
- [IDS Logging on page 88](#)
- [Assessing Bridge Component Versions on page 88](#)
- [Documanage Diagnostic Tests on page 88](#)
- [Distinguishing Bridge Problems From Documanage Problems on page 88](#)
- [Distinguishing Bridge Server from Bridge Client Problems on page 89](#)

Bridge Logging

Documanager Bridge provides an extensive diagnostic log. This is usually your first reference when you need to troubleshoot functional problems. See the description of the [Logging] INI options for details on how to configure the Bridge Log. Since it reduces performance, you will ordinarily run with the Bridge Log turned off, but you will want to turn it on and provide a copy of it to Documanager Support personnel if there is a problem. Note that there are separate [Logging] control groups for the Bridge server rules, configured in `dmg_brs.ini`, and the Bridge client rules, configured in `dmg_brc.ini`.

`DmgBrsInit` writes errors and warnings to the log if it encounters problems during Bridge initialization, which `DmgBrsInit` always logs in some detail even when Bridge Logging is turned off. In particular, you can refer to `DmgBrsInit`'s logging to see warnings about out of date components.

Errors and exceptions are emphasized in the log with two asterisks. Warnings are emphasized with the word `WARNING` in all caps.

When studying the performance of Bridge rules, you will ordinarily to turn off Bridge Logging, since Bridge Logging can have a large effect on rule performance. However, there are times when you'll use it to see the relative lengths of time required by various operations with a single rule.

IDS Logging

See the documentation on the IDS [Debug] INI control group for information on a very powerful tool for tracking the execution of rules and their performance. When studying the performance of Bridge rules, be sure to turn off Bridge Logging, since Bridge Logging can have a large effect on rule performance.

Assessing Bridge Component Versions

To assess whether the components of the Bridge are up to date, you can refer to the start of the Bridge log (see Bridge Logging above). See also the rules `DmgBrcShowVersion` and `DmgBrsShowVersions`.

Documanager Diagnostic Tests

The Bridge can be configured to run Documanager Diagnostic Tests. For more information on Documanager Diagnostic Tests, see the Documanager documentation. For more information on configuring the Bridge to run these tests, see the descriptions of the rule `DmgBrsRunDocumanagerDiagnostic` and of the INI options for [DmgDiagnostics].

Distinguishing Bridge Problems From Documanager Problems

It is valuable to be able to tell if the source of some particular problem is in the Bridge or the way it is configured, or if the source is in Documanager.

One approach is to study and compare the Bridge log and the Documanager log for the time during which the problem was apparent.

Another very helpful approach is to see if carrying out an operation in the Documanager Workstation that is similar to the operation in the Bridge causes the same problem. If it does, the problem is probably in Documanager. For example, if you search a Cabinet with the Bridge and find no Documents in it when you expect to find Documents, try opening the same Cabinet with the Documanager Workstation to see if the Documents are visible.

Distinguishing Bridge Server from Bridge Client Problems

It is important to be able to tell if the source of a problem is due to the way the Bridge server is configured or the Bridge server rules, or to the Bridge client application.

The primary tool for distinguishing server problems from client problems is the Bridge server log. Examine the log for the request in question to see if the input attachment variables and their values are correct, to see if any warnings or errors have been posted to the log, and to see if the output attachment variables and their values are correct. Also, check that the Bridge server INI options have been configured as desired. If all of these things appear to be in order, then the problem is likely to be with the Bridge client application, and further troubleshooting should focus on the client.

