

webMethods Lotus Notes Adapter Installation and User's Guide

Version 6.0

July 2012

This document applies to webMethods Lotus Notes Adapter 6.0 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2005-2022 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <https://softwareag.com/licenses/>.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at <https://softwareag.com/licenses/> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <https://softwareag.com/licenses/> and/or in the root installation directory of the licensed product(s).

Document ID: ADAPTER-LNS-IUG-60-20221202

Table of Contents

About this Guide	5
Document Conventions.....	6
Online Information and Support.....	7
Data Protection.....	8
1 Overview of the Lotus Notes Adapter	9
About the Lotus Notes Adapter.....	10
Architecture and Components.....	10
Package Management.....	12
Adapter Connections.....	13
Adapter Services.....	14
Lotus Notes-to-webMethods Communications.....	17
Using Version Control Systems to Manage Adapter Elements.....	18
Optimize Infrastructure Data Collector Support for webMethods Lotus Notes Adapter....	18
Viewing the Adapter's Update Level.....	19
Controlling Pagination.....	19
2 Installing webMethods Lotus Notes Adapter 6.0	21
Overview.....	22
Requirements.....	22
Installing Lotus Notes Adapter 6.0.....	22
Upgrading to Lotus Notes Adapter 6.0.....	23
Uninstalling Lotus Notes Adapter 6.0.....	25
3 Package Management	27
Overview.....	28
Lotus Notes Adapter Package Management.....	28
Group Access Control.....	31
Lotus Notes Adapter in a Clustered Environment.....	32
4 Adapter Connections	35
Overview.....	36
Before Configuring or Managing Adapter Connections.....	36
Configuring Lotus Notes Adapter Connections.....	36
Dynamically Changing a Service's Connection at Run Time.....	39
Viewing Adapter Connection Parameters.....	40
Editing Adapter Connections.....	41
Copying Adapter Connections.....	42
Deleting Adapter Connections.....	42
Enabling Adapter Connections.....	43
Disabling Adapter Connections.....	43

5 Adapter Services.....	45
Overview.....	46
Before Configuring or Managing Adapter Services.....	46
Using Lotus Notes Adapter Services.....	46
Configuring Lotus Domino Operation Services.....	49
Testing Adapter Services.....	50
Viewing Adapter Services.....	51
Editing Adapter Services.....	52
Deleting Adapter Services.....	52
Enabling Automatic Data Validation.....	53
Reloading Adapter Values.....	54
Reloading for a Single Adapter Service.....	54
Reloading for All Adapter Services.....	54
Editing the XML Data File.....	55
6 Lotus Notes-to-webMethods Communications.....	57
Overview.....	58
Setting Up the Domino Server for Outbound Transactions.....	58
Using SSL to Communicate with Integration Server.....	61
7 Predefined Health Indicator.....	63
Predefined Health Indicator.....	64
8 Administrator APIs.....	65
Administrator APIs.....	66
9 Configuration Variables Templates for Adapter Assets in Microservices Runtime.....	67
Configuration Variables Templates for Adapter Assets in Microservices Runtime.....	68
10 Logging and Exception Handling.....	69
Overview.....	70
Lotus Notes Adapter Logging Levels.....	70
Lotus Notes Adapter Message Logging.....	71
Handling Lotus Notes Exceptions.....	72
Lotus Notes Adapter Error Codes.....	72
A Built-In Services Reference.....	77
Overview.....	78
Domino Operations Folder Structure.....	78
Summary of Built-In Services.....	79
Built-In Services Reference.....	88

About this Guide

■ Document Conventions	6
■ Online Information and Support	7
■ Data Protection	8

This guide describes how to install, configure, and use webMethods Lotus Notes Adapter Version 6.0. It contains information for administrators who configure and manage a webMethods system and for application developers who want to create webMethods Integration Server services that exchange data with Lotus Notes systems.

To use this guide effectively, you should be familiar with:

- Lotus Notes and how your organization uses it
- The terminology and basic operations of your operating system
- The setup and operation of webMethods Integration Server.
- How to perform basic tasks with Software AG Designer.

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Narrowfont	Identifies service names and locations in the format <i>folder.subfolder.service</i> , APIs, Java classes, methods, properties.
<i>Italic</i>	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
Monospace font	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the symbol.
[]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Product Documentation

You can find the product documentation on our documentation website at <https://documentation.softwareag.com>.

In addition, you can also access the cloud product documentation via <https://www.softwareag.cloud>. Navigate to the desired product and then, depending on your solution, go to “Developer Center”, “User Center” or “Documentation”.

Product Training

You can find helpful product training material on our Learning Portal at <https://knowledge.softwareag.com>.

Tech Community

You can collaborate with Software AG experts on our Tech Community website at <https://techcommunity.softwareag.com>. From here you can, for example:

- Browse through our vast knowledge base.
- Ask questions and find answers in our discussion forums.
- Get the latest Software AG news and announcements.
- Explore our communities.
- Go to our public GitHub and Docker repositories at <https://github.com/softwareag> and <https://hub.docker.com/u/softwareag> and discover additional Software AG resources.

Product Support

Support for Software AG products is provided to licensed customers via our Empower Portal at <https://empower.softwareag.com>. Many services on this portal require that you have an account. If you do not yet have one, you can request it at <https://empower.softwareag.com/register>. Once you have an account, you can, for example:

- Download products, updates and fixes.
- Search the Knowledge Center for technical information and tips.
- Subscribe to early warnings and critical alerts.
- Open and update support incidents.
- Add product feature requests.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

1 Overview of the Lotus Notes Adapter

■ About the Lotus Notes Adapter	10
■ Architecture and Components	10
■ Package Management	12
■ Adapter Connections	13
■ Adapter Services	14
■ Lotus Notes-to-webMethods Communications	17
■ Using Version Control Systems to Manage Adapter Elements	18
■ Optimize Infrastructure Data Collector Support for webMethods Lotus Notes Adapter ..	18
■ Viewing the Adapter's Update Level	19
■ Controlling Pagination	19

About the Lotus Notes Adapter

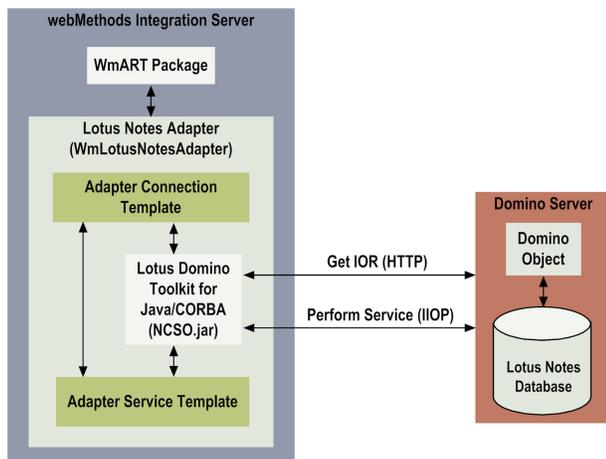
webMethods Lotus Notes Adapter is an add-on to webMethods Integration Server that enables you to exchange data with Lotus Domino Servers. Using Lotus Notes Adapter, webMethods Integration Server developers can create and run services that execute transactions to retrieve data from, and insert and update data in, Domino Servers. For example, you can use Lotus Notes Adapter to retrieve a document from the Domino Server based on data from another system connected to Integration Server.

Lotus Notes applications may also act as clients to Integration Server. A code sample for setting up this functionality is provided.

Architecture and Components

Lotus Notes Adapter provides a set of user interfaces, services, and templates that enable you to create integrations with Lotus Notes systems. The adapter is provided as a single package that must be installed on Integration Server. For detailed installation instructions, see [“Installing webMethods Lotus Notes Adapter 6.0” on page 21](#). For software requirements, see *webMethods Adapters System Requirements*.

The following diagram shows at a high level how Lotus Notes Adapter components interface with a Domino Server using the Lotus Domino Toolkit for Java/CORBA to access remote Domino objects.



- **webMethods Integration Server** . Lotus Notes Adapter is installed and runs on Integration Server.
- **WmART Package**. The WmART package provides a common framework for webMethods 6 and later adapters to use Integration Server functionality, making Integration Server the run-time environment for Lotus Notes Adapter. The WmART package is installed with Integration Server.
- **Lotus Notes Adapter** . Lotus Notes Adapter is delivered as a single package called WmLotusNotesAdapter. The adapter provides Integration Server Administrator user interfaces that enable you to configure and manage adapter connections, and user interfaces in Software

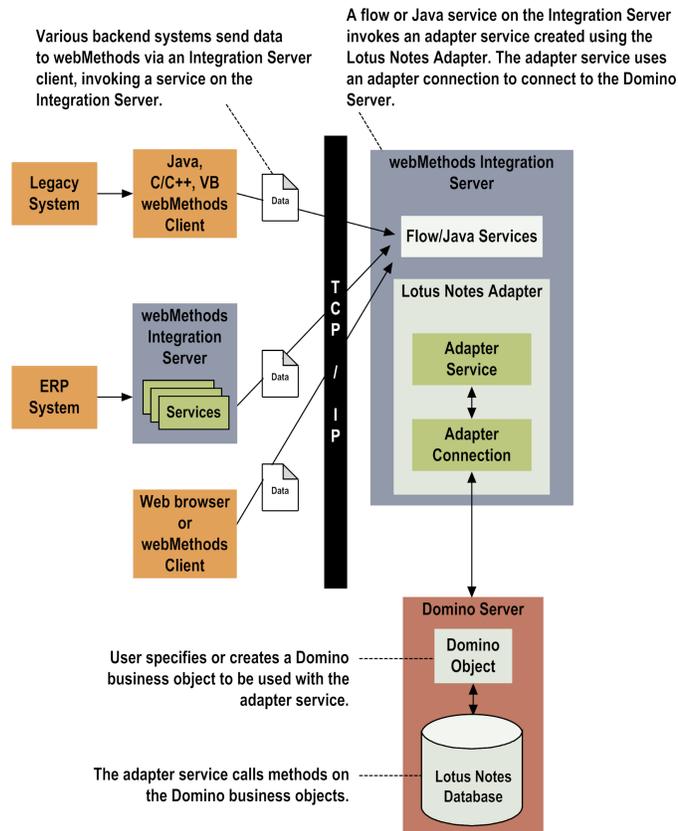
AG Designer that enable you to configure and manage adapter services. Lotus Notes Adapter installation includes a single template from which you can create all Lotus Notes Adapter services.

- **Adapter Connection Template.** Lotus Notes Adapter connections enable you to create adapter services. You must configure an adapter connection before you can create adapter services. The adapter provides a template for adapter connections in the Integration Server Administrator. For a detailed description of adapter connections, see [“Adapter Connections” on page 13](#).
- **Adapter Service Template.** Lotus Notes Adapter provides an adapter service template that enables Integration Server to communicate with Domino Servers. Lotus Notes Adapter’s adapter service template is available in Designer. Services created from this template allow you to manipulate Domino objects based on available methods in the IBM Lotus Domino Toolkit for Java/CORBA. For more information about adapter services and using the adapter service template, see [“Adapter Services” on page 14](#).
- **Lotus Domino Toolkit for Java/CORBA:** Lotus Notes Adapter communicates with the Domino Server using the Lotus Domino Toolkit for Java/CORBA (NCSO.jar or NCSOW.jar). The remote Domino object classes in the NCSO.jar and NCSOW.jar files use CORBA to access the Domino Server over a TCP/IP network. Remote access consists of two parts:
 - Lotus Notes Adapter receives the Domino Server's initial object as an IOR (Interoperable Object Reference) via the HTTP protocol.
 - Lotus Notes Adapter receives additional objects and makes method calls over IIOP.

For more information about the Lotus Domino Toolkit for Java/CORBA and the NCSO.jar or NCSOW.jar files that the adapter requires, see *webMethods Adapters System Requirements* and IBM's *Java Access to Domino Objects - IBM Redbook Library Part 1*. This information is also available in the documentation for the Lotus Domino Toolkit for Java/CORBA and on the IBM Lotus web site.

For a detailed description of the run-time data flow for Lotus Notes Adapter services, see [“Adapter Service Run-Time Processing” on page 16](#).

The following diagram illustrates a typical business-process integration at run time involving Lotus Notes Adapter and the Domino Server.



Package Management

Lotus Notes Adapter is provided as a package called `WmLotusNotesAdapter` that you manage like any package on Integration Server. There are several considerations regarding how you set up and effectively manage your packages on Integration Server, such as those described in the following list.

- Create user-defined packages for your connections, and adapter services. See [“Package Management” on page 27](#) for details.
- Understand how package dependencies work so you make the best decisions regarding how you manage your adapter services. See [“Package Dependency Requirements and Guidelines” on page 29](#) for details.
- Control which development groups have access to which adapter services. See [“Group Access Control” on page 31](#) for details.
- Understand how clustering, an advanced feature of Integration Server, works to effectively manage your adapter services. See [“Lotus Notes Adapter in a Clustered Environment” on page 32](#) for details.
- Enable and disable packages. See [“Enabling Packages” on page 29](#) and [“Disabling Packages” on page 30](#) for details.

- Load, reload, and unload packages. See [“Loading, Reloading, and Unloading Packages”](#) on page 30.

Adapter Connections

Lotus Notes Adapter uses an adapter connection to create a Lotus Domino session object (`lotus.domino.Session`). A session object is required before doing any work with a Domino Server. Lotus Notes Adapter provides one connection type, Lotus Notes Connection. A connection for Lotus Notes Adapter is a placeholder for connection information: user, password, host, and an HTTP/HTTPS option. More than one session object can be created with an adapter connection.

A connection does not automatically start or maintain connections to a Domino Server; it only creates session objects to a Domino Server. And because adapter connections do not store session objects created to a Domino Server, there is no connection pooling support.

Integration developers using Lotus Notes Adapter may need to create session objects for many users, sometimes hundreds. Creating an adapter connection for each of these users is a significant administrative effort. Developers can create session objects by configuring an adapter service to use a certain adapter connection and optionally override the user and password in the connection. That is, the credentials stored in the connection are the default values when no credentials are specified when the connection creates a session object. This enables one connection to create session objects for more than one user. For more information about creating Domino session objects, see the Lotus Domino Toolkit for Java/CORBA documentation. There is no transactional support for an adapter connection because the Lotus Domino Toolkit for Java/CORBA does not provide it.

Integration developers are responsible for recycling any Domino objects (especially session objects) they create locally to free local and remote resources. The adapter provides two utility services, `wm.in.utils.garbage:recycle` and `wm.in.authentication:logout`, for this purpose. For more information about these services, see [“Freeing Resources After Processing”](#) on page 16.

You configure connections using the Integration Server Administrator. You must have Integration Server administrator privileges to access the Lotus Notes Adapter's administrative screens.

For instructions for configuring, viewing, editing, enabling, and disabling Lotus Notes Adapter connections, see [“Adapter Connections”](#) on page 35. For information about setting user privileges, see the *webMethods Integration Server Administrator's Guide* for your release.

For a list of tasks that you must do before you can create your connections, see [“Before Configuring or Managing Adapter Connections”](#) on page 36.

Built-In Services For Connections

Integration Server provides built-in services that enable you to programmatically control connections. You can use them to enable and disable a connection, and to return usage statistics and the current state (Enabled or Disabled) and error status for a connection. These services are located in the WmART package, in the `pub.art.connection` folder.

The built-in service `pub.art.service:setAdapterServiceNodeConnection` enables you to change the connection associated with an adapter service. For information, see [“Changing the Connection Associated With an Adapter Service at Design Time”](#) on page 14.

For details, see the *webMethods Integration Server Built-In Services Reference* for your release.

Changing the Connection Associated With an Adapter Service at Design Time

Integration Server provides a built-in service that you can use at design time to change the connection associated with an adapter service. This built-in service, named `setAdapterServiceNodeConnection`, is provided in the WmART package's `pub.art.service` folder. Using this service, you can change the specific connection associated with an adapter service at design time so that you do not need to create and maintain multiple adapter services.

Note:

These built-in services can be run at design time only; do not use them within an Integration Server flow or Java service. You must run this service directly from Designer by selecting and running the service.

For details, see the *webMethods Integration Server Built-In Services Reference* for your release.

Other built-in services enable you to control connections; for more information, see [“Built-In Services For Connections” on page 13](#).

Changing the Connection Associated with an Adapter Service at Run Time

Integration Server enables you to dynamically select the connection a service uses to interact with the adapter's resource. This feature enables one service to interact with multiple, similar backend resources.

For example, a service can be defined to use a default connection that interacts with your company's North American Lotus Domino server. However, at run time you can override the default connection and instead use another connection to interact with the company's European Lotus Domino server.

For more information about overriding a service's default connection at runtime, see [“Dynamically Changing a Service's Connection at Run Time” on page 39](#).

Adapter Services

Adapter services allow you to connect to, and initiate operations on, a Domino Server. Adapter services are based on a template provided with Lotus Notes Adapter. With Lotus Notes Adapter, an adapter service wraps a method you select from the adapter service template. These methods belong to the Java `lotus.domino` classes supplied by the Lotus Domino Toolkit for Java/CORBA, and are used to remotely execute methods on Domino objects. Lotus Notes Adapter supports a subset of the classes in the toolkit. The methods in the template are defined in an XML data file. For information about managing the XML data file, see [“Editing the XML Data File” on page 55](#).

Lotus Notes Adapter provides adapter services that have been pre-configured to call each of the supported methods in the Java `lotus.domino` classes. These adapter services use a preconfigured

adapter connection that serves as a placeholder for connection information. Also provided with Lotus Notes Adapter are built-in flow and Java services that wrap one or more of the adapter services. Integration developers use the flow and Java services in their integrations instead of calling the adapter services directly.

For information about the adapter services and how they are provided in Designer, see [“Built-In Services Reference” on page 77](#). For information about the classes and methods provided in `lotus.domino` package, see the Lotus Domino Toolkit for Java/CORBA documentation.

Note: A samples package is available that contains adapter services that show how to work with the Domino objects. You can use the samples as "working examples" from which you can build and test your own integrations. To download Lotus Notes Adapter, go to the [Empower Product Support Web site](#).

Using the Adapter Service Template

You use Designer to configure an adapter service. For more information, see the *webMethods Service Development Help* for your release.

Configuring a new adapter service from the adapter service template is straightforward. Using Designer, you assign the service a default adapter connection, select the adapter service template, select the class and method to execute on the Domino Server, and specify how the adapter is to generate the input and output signatures.

Note:

There can be more than one method in the class with the same name. The adapter distinguishes them by appending the method signatures to the method name.

Configure at least one adapter service that executes the `createSession` method in the `lotus.domino.Session` class, which creates the connection object for the Domino Server. Then configure additional adapter services as needed to perform specific Domino operations. There is no requirement to associate the same connection to adapter services after a session object has been created.

After you configure an adapter service, you can incorporate it into a flow service or Java service to interact with a Lotus Notes system. For example, you can create a flow service that creates an e-mail and sends it using the Lotus Notes system.

Important:

All services must first invoke an adapter service that specifies the `lotus.domino.Session` class and `createSession` method to create the connection object for the Domino Server. For information about creating this adapter service, see [“Example Lotus Notes Operation” on page 47](#).

Lotus Notes Adapter provides the following adapter service template:

Adapter Service Template	Description
Domino Operation	Provides classes and methods for Domino objects.

Adapter Service Template	Description
	See “Configuring Lotus Domino Operation Services” on page 49 for instructions.

The input and output signatures for the adapter service are shown in the service template, and are generated as Java data primitives when you create the service. However, you can force the signatures to be generated using Strings by specifying the **Primitive** option.

When you run the adapter service, specify the correct, valid inputs. For information about the inputs, see the Lotus Domino Toolkit for Java/CORBA documentation. Note that in the toolkit documentation, "" is referred to as "null". Lotus Notes Adapter converts nulls to "".

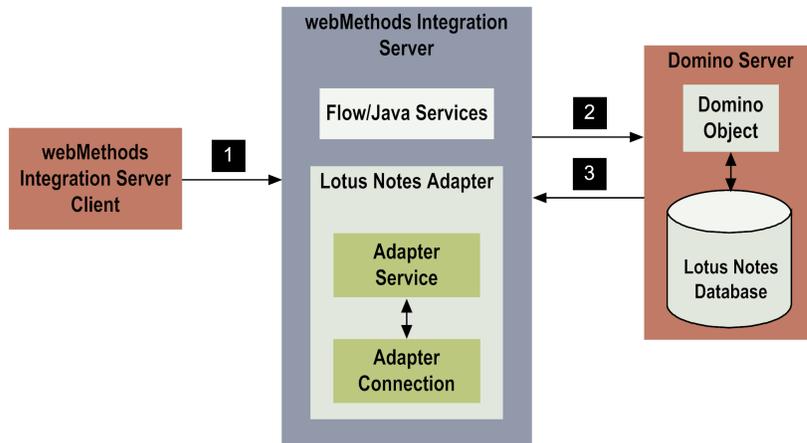
Freeing Resources After Processing

To free local and remote resources created by the services, call the `wm.ln.utils.garbage:recycle` and/or `wm.ln.authentication:logout` utility service in each flow or Java service. For example, within a flow you create a session object, send an email, call `recycle`, then send a newsletter, then call `logout`.

- The `wm.ln.utils.garbage:recycle` service recycles all Domino objects at the root level of the pipeline, including session objects if they exist. Session objects are recycled last. Use the `recycle` service between processing steps if you have more work to do and want to reuse the same session object. Do leave the session object at the root level of the pipeline or else it will be recycled. For more information, see [“wm.ln.utils.garbage:recycle” on page 160](#).
- The `wm.ln.authentication:logout` service checks to see if there is a session object at the root level of pipeline. The `logout` service requires that there be a session object in the pipeline, and if there is, it scans the pipeline, and then call the `recycle` service to recycle all other objects, and then recycles the session object. If there is no session object, the `logout` service fails. Always use the `logout` service as the last step in a flow or Java service to ensure that all objects are recycled. For information about the input and output parameters, see [“wm.ln.authentication:logout” on page 101](#).

Adapter Service Run-Time Processing

The following diagram illustrates how Lotus Notes Adapter processes adapter services at run time.



Step	Description
1	An Integration Server client, typically using a flow or Java service, invokes a Lotus Notes Adapter service on the webMethods Integration Server to perform an operation on a Domino Server. You configured the adapter service earlier using Designer.
2	The adapter service uses the methods and classes in the Lotus Domino Toolkit for Java/CORBA to perform the necessary operations.
3	If the operation is successful, the adapter service returns the output from the operation, if applicable. If the operation is unsuccessful, the adapter service returns an error. For more information about how the adapter handles exceptions, see “Built-In Services Reference” on page 77 .

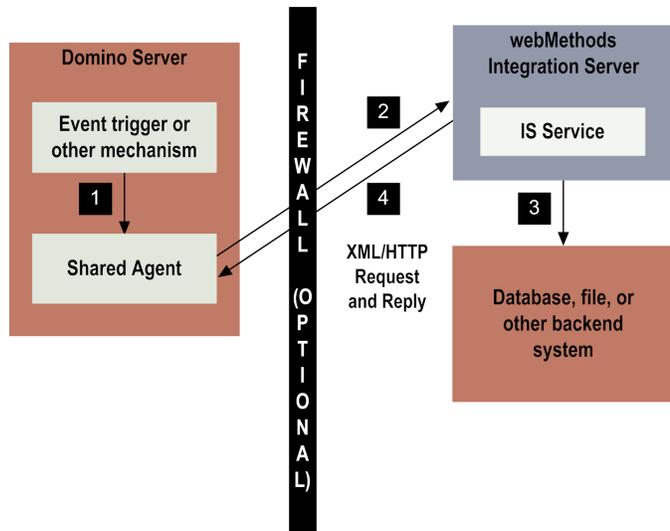
Lotus Notes-to-webMethods Communications

For Lotus Notes-to-webMethods (outbound) communications, you must write a shared agent in Java or LotusScript that invokes a service on the specified webMethods Integration Server. For more information, see [“Lotus Notes-to-webMethods Communications” on page 57](#).

Lotus Notes-to-webMethods Run-Time Processing

Integration Server processes service requests from an agent running on a Domino Server. You must understand how agents run on a Domino server. For more information about using webMethods clients, see the *webMethods Service Development Help* for your release.

Lotus Notes Adapter processes Lotus Notes-to-webMethods transactions at run time as follows:



Step	Description
1	Data inside the Lotus Notes system is added, deleted, or modified, or another type of event occurs that triggers the shared agent for Lotus Notes-to-webMethods communications.
2	The shared agent connects to Integration Server.
3	One or more services execute based on the data sent by the shared agent.
4	Integration Server synchronously returns the result to the shared agent.

Using Version Control Systems to Manage Adapter Elements

The adapter supports the Version Control System (VCS) Integration feature provided by Designer. When you enable the feature in Integration Server, you can check adapter packages or elements into and out of your version control system from Designer. For more information about the VCS Integration feature, see the *Configuring the VCS Integration Feature*.

Beginning with Integration Server 8.2 SP3, the adapter supports the local service development feature in Designer. This feature extends the functionality of the VCS Integration feature to check package elements and their supporting files into and out of a VCS directly from Designer. For more information about local service development and how it compares to the VCS Integration feature, see the *webMethods Service Development Help*.

Optimize Infrastructure Data Collector Support for webMethods Lotus Notes Adapter

Optimize Infrastructure Data Collector monitors the system and operational data associated with webMethods run-time components such as Integration Servers, Broker Servers, Brokers, and

adapters, and reports the status of these components on Optimize for Infrastructure or other external tools. When you start monitoring an Integration Server, Infrastructure Data Collector automatically starts monitoring all ART-based adapters that are installed on Integration Server.

For information about monitored key performance indicators (KPIs) collected for the monitored adapter components, see the *webMethods Optimize User's Guide* and *Administering webMethods Optimize* for your release.

Viewing the Adapter's Update Level

You can view the list of updates that have been applied to the adapter. The list of updates appears in the **Updates** field on the adapter's About page in the Integration Server Administrator.

Controlling Pagination

When using the adapter on Integration Server 8.0 and later, you can control the number of items that are displayed on the adapter Connections screen and Notifications screen. By default, 10 items are displayed per page. Click **Next** and **Previous** to move through the pages, or click a page number to go directly to a page.

To change the number of items displayed per page, set the `watt.art.page.size` property and specify a different number of items.

› To set the number of items per page

1. From Integration Server Administrator, click **Settings > Extended**.
2. Click **Edit Extended Settings**. In the Extended Settings editor, add or update the `watt.art.page.size` property to specify the preferred number of items to display per page. For example, to display 50 items per page, specify:

```
watt.art.page.size=50
```

3. Click **Save Changes**. The property appears in the Extended Settings list.
4. For more information about working with extended configuration settings, see the *webMethods Integration Server Administrator's Guide* for your release.

2 Installing webMethods Lotus Notes Adapter 6.0

■ Overview	22
■ Requirements	22
■ Installing Lotus Notes Adapter 6.0	22
■ Upgrading to Lotus Notes Adapter 6.0	23
■ Uninstalling Lotus Notes Adapter 6.0	25

Overview

This chapter explains how to install, upgrade, and uninstall webMethods Lotus Notes Adapter 6.0. The instructions use the Software AG Installer and the Software AG Uninstaller wizards. For complete information about the wizards or other installation methods, or to install other webMethods products, see *Installing webMethods Products On Premises* for your release.

Requirements

For a list of the operating systems, Lotus Domino Server products, and webMethods products supported by Lotus Notes Adapter 6.0, see *webMethods Adapters System Requirements* .

Lotus Notes Adapter 6.0 has no hardware requirements beyond those of its host Integration Server.

Installing Lotus Notes Adapter 6.0

To install Lotus Notes Adapter on AS/400, you must use the Installer's console mode, described in the *Installing webMethods Products On Premises* for your release.

Note:

If you are installing the adapter in a clustered environment, you must install it on each Integration Server in the cluster, and each installation must be identical. For more information about working with the adapter in a clustered environment, see “ [Lotus Notes Adapter in a Clustered Environment](#)” on page 32.

» To install Lotus Notes Adapter 6.0

1. Download Installer from the [Empower Product Support Web site](#).
2. If you are installing the adapter on an existing Integration Server, shut down Integration Server.
3. Start the Installer wizard.
4. Choose the webMethods release that includes the Integration Server on which to install the adapter. For example, if you want to install the adapter on Integration Server 9.12, choose the 9.12 release.
5. Specify the installation directory as follows:
 - If you are installing on an existing Integration Server, specify the webMethods installation directory that contains the host Integration Server.
 - If you are installing both the host Integration Server and the adapter, specify the installation directory to use.

The Installer will install the adapter in the *Integration Server_directory* \packages directory.

6. In the product selection list, select **Adapters > webMethods Lotus Notes Adapter 6.0**. You can also choose to install documentation.
7. After installation is complete, configure the Domino Server to accept transactions from Integration Server. To do so, configure the HTTP and IIOP tasks on the Domino Server, make sure the tasks are running, and make sure the tasks are listed in the `notes.ini` file in the Domino Server (for example, `ServerTasks=task1,taskn,http,iiop`). For instructions, see your Domino Server administration documentation.
8. Set up the Lotus Notes Adapter environment. To enable the Lotus Notes Adapter to communicate with the Lotus Domino objects on the Domino Server, the adapter requires access to `NCSO.jar`, which is provided with the Lotus Domino Toolkit for Java/CORBA.
 - a. Go to the *Integration Server_directory* `\instances\instance_name\packages\WmLotusNotesAdapter\code` directory.
 - b. Create a `jars\static` directory.
 - c. Copy the `NCSO.jar` or `NCSOW.jar` file to the `jars\static` directory. The file is located in the directory where you installed the Domino Server (for example, `LotusDomino_directory\data\domino\java`).
9. Set up the Lotus Notes Adapter environment for Integration Server 10.11 onwards.

The connections are not enabled as Integration Server 10.11 uses Java 11. The Lotus domino client uses IIOP (Internet Inter-ORB Protocol) to communicate with the Domino server. However, the CORBA modules (OMG CORBA API, an ORB implementation) is removed from Java 11 on Java SE Platform.

- a. Download the GlassFish ORB implementation `glassfish-corba-omgapi-4.2.3.jar` from Maven
`https://mvnrepository.com/artifact/org.glassfish.corba/glassfish-corba-omgapi/4.2.3`.
- b. Copy the JAR file to the *Integration Server_directory* `\instances\instance_name\packages\WmLotusNotesAdapter\code\jars` directory.
- c. Restart Integration Server if already started.

Upgrading to Lotus Notes Adapter 6.0

You can upgrade to Lotus Notes Adapter 6.0 from Lotus Notes Adapter 1.0.x.

➤ To upgrade from Lotus Notes Adapter 1.0.x

1. Alias management and other services that were available with Lotus Notes Adapter 1.0.x are not available in Lotus Notes Adapter 6.0 (see [“Unavailable Services and Records” on page 25](#) for a complete list). If your Lotus Notes Adapter 1.0.x integrations use any services or records

that are no longer available, disable or delete the corresponding flow steps in your integrations. If necessary, replace the functionality with other flow or Java services.

If you are not sure whether your integrations use any of the unavailable services or records, use the **Find Dependents** tool in Designer.

For instructions on this step, see the *webMethods Service Development Help* for your release.

2. Create a list of each alias you currently use in Lotus Notes Adapter 1.0.x. Include all information about the alias, including user, password, and host.
3. In Integration Server Administrator for Lotus Notes Adapter 1.0.x, disable the LotusNotes 1.0.x package. Do not disable the user-defined package that contains your integrations. For instructions, see the *webMethods Integration Server Administrator's Guide* for your release.
4. Install Lotus Notes Adapter 6.0 using the instructions in ["Installing Lotus Notes Adapter 6.0" on page 22](#).
5. In Integration Server Administrator for Lotus Notes Adapter 6.0, configure a Lotus Notes Adapter 6.0 connection for each alias you recorded in step 2. For instructions, see ["Adapter Connections" on page 35](#).
6. In Designer, do the following:
 - a. Configure an adapter service that executes the createSession method in the lotus.domino.Session class for each connection object you created in step 5. For instructions, see ["Adapter Services" on page 45](#).
 - b. Modify the wm.ln.authentication:login service in the WmLotusNotesAdapter package, as follows:
 - If you only have one alias, modify the wm.ln.authentication:login service to call the adapter service you created in step 6a.
 - If you have more than one alias, modify the signature of the wm.ln.authentication:login service to have an alias string as input. Based on the value of this string, the service should call the createSession adapter service that is associated with the corresponding connection for the alias provided.

In either case, make sure the login service leaves a session object at the root level of the pipeline.

7. Perform acceptance testing to make sure the integrations, adapter connections, and adapter services are functioning correctly.

Unavailable Services and Records

The following services that were available with Lotus Notes Adapter 1.0.x are not available in Lotus Notes Adapter 6.0:

- `wm.In.mimeEntity:transformXML`
- `wm.In.utils.alias:add`
- `wm.In.utils.alias:del`
- `wm.In.utils.alias:delAlias`
- `wm.In.utils.alias:edit`
- `wm.In.utils.alias:editAlias`
- `wm.In.utils.alias:get`
- `wm.In.utils.alias:getAlias`
- `wm.In.utils.alias:getAliasList`
- `wm.In.utils.alias:getCnfFile`
- `wm.In.utils.alias:saveAliasSettings`
- `wm.In.utils.alias:saveCnfFile`

The following record is no longer available in Lotus Notes Adapter 6.0:

- `wm.In.utils.alias.recs:alias`

For a list of all services provided with Lotus Notes Adapter 6.0, see [“Built-In Services Reference” on page 77](#).

Uninstalling Lotus Notes Adapter 6.0

➤ To uninstall Lotus Notes Adapter 6.0

1. Shut down the host Integration Server. You do not need to shut down any other webMethods products or applications that are running on your machine.
2. Start Software AG Uninstaller, selecting the webMethods installation directory that contains the host Integration Server. In the product selection list, select **Adapters > webMethods Lotus Notes Adapter 6.0**. You can also choose to uninstall documentation.
3. Restart the host Integration Server.
4. Uninstaller removes all Lotus Notes Adapter 6.0-related files that were installed. However, Uninstaller does not delete files created after you installed the adapter (for example, user-created

or configuration files), nor does it delete the adapter directory structure. You can go to the *Integration Server_directory* \packages directory and delete the adapter directory.

3 Package Management

■ Overview	28
■ Lotus Notes Adapter Package Management	28
■ Group Access Control	31
■ Lotus Notes Adapter in a Clustered Environment	32

Overview

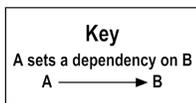
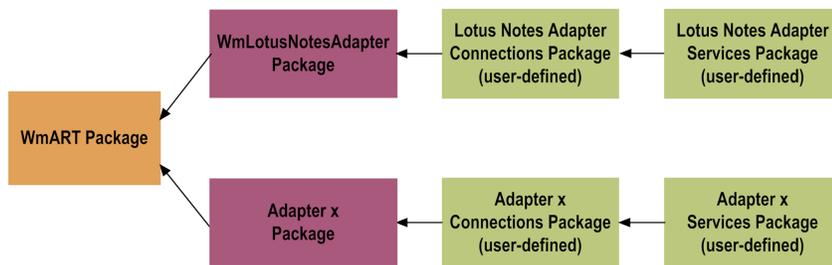
The following sections describe how to set up and manage your Lotus Notes Adapter packages, set up Access Control Lists (ACL), and use the adapter in a clustered environment.

Lotus Notes Adapter Package Management

Lotus Notes Adapter is provided as a package called WmLotusNotesAdapter. You manage the WmLotusNotesAdapter package as you would manage any package on Integration Server.

When you create connections and adapter services, define them in user-defined packages rather than in the WmLotusNotesAdapter package. Doing so will allow you to manage the package more easily.

As you create user-defined packages in which to store connection and adapter services, use the package management functionality provided in the Designer and set the user-defined packages to have a dependency on the WmLotusNotesAdapter package. That way, when the WmLotusNotesAdapter package loads or reloads, the user-defined packages load automatically. The following diagram illustrates Lotus Notes Adapter package management.



Package management tasks include:

- [Setting package dependencies \(see “Package Dependency Requirements and Guidelines” on page 29\)](#)
- [“Enabling Packages” on page 29](#)
- [“Disabling Packages” on page 30](#)
- [“Group Access Control” on page 31](#)

Package Dependency Requirements and Guidelines

This section contains a list of dependency requirements and guidelines for user-defined packages. For instructions for setting package dependencies, see the *webMethods Service Development Help* for your release.

- A user-defined package must have a dependency on its associated adapter package, WmLotusNotesAdapter. (The WmLotusNotesAdapter package has a dependency on the WmART package.)
- Package dependencies ensure that at startup Integration Server automatically loads or reloads all packages in the proper order: the WmART package first, the adapter package next, and the user-defined packages last. The WmART package is automatically installed when you install Integration Server. You should not need to manually reload the WmART package.
- If the connections and adapter services of an adapter are defined in different packages:
 - A package that contains the connection(s) must have a dependency on the adapter package.
 - Packages that contain adapter services must have a dependency on their associated connection package.
- Keep connections for different adapters in separate packages so that you do not create interdependencies between adapters. If a package contains connections for two different adapters, and you reload one of the adapter packages, the connections for both adapters will reload automatically.
- Integration Server will not allow you to enable a package if it has a dependency on another package that is disabled. That is, before you can enable your package, you must enable all packages on which your package depends. For information about enabling packages, see [“Enabling Packages” on page 29](#).
- Integration Server will allow you to disable a package even if another package that is enabled has a dependency on it. Therefore, you must manually disable any user-defined packages that have a dependency on the adapter package before you disable the adapter package. For information about disabling packages, see [“Disabling Packages” on page 30](#).
- You can name connections and adapter services the same name provided that they are in different folders and packages.

Enabling Packages

All packages are automatically enabled by default.

➤ To enable a package

1. Open the Integration Server Administrator if it is not already open.
2. In the **Packages** menu of the navigation area, click **Management**.

3. Click **No** in the **Enabled** column. The server displays a ✓ and **Yes** in the **Enabled** column.

Note:

Enabling an adapter package will not cause its associated user-defined package(s) to be reloaded. For information about reloading packages, see [“Loading, Reloading, and Unloading Packages”](#) on page 30.

Important:

Before you manually enable a user-defined package, you must first enable its associated adapter package (WmLotusNotesAdapter). Similarly, if your adapter has multiple user-defined packages, and you want to disable some of them, disable the adapter package first. Otherwise, errors will be issued when you try to access the remaining enabled user-defined packages.

Disabling Packages

When you want to temporarily prohibit access to the elements in a package, disable the package. When you disable a package, the server unloads all of its elements from memory. Disabling a package prevents the Integration Server Administrator from loading that package at startup. A disabled package will remain disabled until you explicitly enable it using the Integration Server Administrator.

➤ To disable a package

1. Open the Integration Server Administrator if it is not already open.
2. In the **Packages** menu of the navigation area, click **Management**.
3. Click **Yes** in the **Enabled** column for the package that you want to disable. The server issues a prompt to verify that you want to disable the package. Click **OK** to disable the package. When the package is disabled, the server displays **No** in the **Enabled** column.

A disabled adapter will:

- Remain disabled until you explicitly enable it using the Integration Server Administrator.
- Not be listed in Designer.

Loading, Reloading, and Unloading Packages

Recall that if user-defined packages are properly configured with a dependency on the adapter package (as described in [“Package Dependency Requirements and Guidelines”](#) on page 29), at startup Integration Server automatically loads or reloads all packages in the proper order: the WmART package first, the adapter package next, and the node packages last. You should not need to manually reload the WmART package.

Reloading Packages Manually

Reloading a user-defined package will not cause its associated adapter package to be reloaded. You can reload adapter packages and user-defined packages from either the Integration Server Administrator (by clicking the **Reload** icon on the Management window) or from Designer (by right-clicking the package and selecting the **Reload Package** option from the menu).

Unloading Packages

At shutdown, the Integration Server unloads packages in the reverse order in which it loaded them: it unloads the node packages first, the adapter package next, and the WmART package last (assuming the dependencies are correct).

Setting Package Dependencies

You set package dependencies if a given package needs services in another package to load before it can load. For example, any packages you create for Lotus Notes Adapter services should identify the Lotus Notes Adapter package (WmLotusNotesAdapter) as a package dependency because they require services in the WmLotusNotesAdapter to load first. Use the following guidelines:

- Set package dependencies from the adapter service package to the package containing the connection if you configure a connection in one package and the adapter services in another package. That is, the package that contains the connection should load before the adapter service package.

When you set this package dependency, it ensures that if someone disables the connection package and then re-enables it, the adapter services will reload correctly.

- If both the connection and adapter services are in the same package, then no dependencies need to be set.
- In general, packages containing connections should have a dependency set to the adapter package itself. That is, the adapter service package should depend on the adapter connection package, which should depend on the adapter package. Similarly, if the adapter services are in the same package as the connections, the only dependency that you need to set is between the adapter connection package and the adapter package.

For more information about setting package dependencies, see the *webMethods Service Development Help* for your release.

Group Access Control

To control which development group has access to which adapter services, use access control lists (ACLs). You can use ACLs to prevent one development group from inadvertently updating the work of another group, or to allow or deny access to services that are restricted to one group but not to others.

For general information about assigning and managing ACLs, see the *webMethods Service Development Help* for your release.

Lotus Notes Adapter in a Clustered Environment

What is webMethods Integration Server Clustering?

Clustering is an advanced feature of the webMethods product suite that substantially extends the reliability, availability, and scalability of Integration Server. Clustering accomplishes this by providing the infrastructure and tools to deploy multiple Integration Servers as if they were a single virtual server and to deliver applications that leverage that architecture. Because this activity is transparent to the client, clustering makes multiple servers look and behave as one.

For details on Integration Server clustering, see the *webMethods Integration Server Clustering Guide* for your release.

Integration Server 8.2 SP2 and higher supports the caching and clustering functionality provided by Terracotta. Caching and clustering are configured at the Integration Server level and Lotus Notes Adapter uses the caching mechanism that is enabled on Integration Server.

With clustering, you get the following benefits:

- **Load balancing.** This feature, provided automatically when you set up a clustered environment, allows you to spread the workload over several servers, thus improving performance and scalability.
- **Failover support.** Clustering enables you to avoid a single point of failure. If a server cannot handle a request, or becomes unavailable, the request is automatically redirected to another server in the cluster.

Note:webMethods Integration Server clustering redirects HTTP and HTTPS requests, but does not redirect FTP or SMTP requests.

- **Scalability.** You can increase your capacity even further by adding new machines running Integration Server to the cluster.

Configuring Lotus Notes Adapter

When you configure the Lotus Notes Adapter to create adapter services, you must:

- Ensure that each Integration Server in the cluster contains an identical set of packages.
- Disable the redirection capability for certain predefined administrative services.

Replicating Packages to webMethods Integration Servers

Every Integration Server in the cluster should contain an identical set of packages that you define using Lotus Notes Adapter; that is, you should replicate Lotus Notes Adapter services and the connections they use.

To ensure consistency, you should create all packages on one server, and replicate them to the other servers. If you allow different servers to contain different services, you might not derive the

full benefits of clustering. For example, if a client requests a service that resides in only one server, and that server is unavailable, the request cannot be successfully redirected to another server.

For information about replicating packages, see the *webMethods Integration Server Administrator's Guide* for your release.

Disabling the Redirection of Administrative Services

As mentioned in “[What is webMethods Integration Server Clustering?](#)” on page 32, a server that cannot handle a client's service request can automatically redirect the request to another server in the cluster. However, the webMethods Adapter Run Time uses certain predefined administrative services that you should not allow to be redirected. These services are used internally when you configure the adapter. If you allow these services to be redirected, your configuration specifications might be saved on multiple servers, which is an undesirable result. For example, if you create two Lotus Notes Adapter services, one might be stored on one server, while the other one might be stored on another server. Remember that all adapter services must reside on all webMethods Integration Servers in the cluster.

➤ To disable the redirection of administrative services

1. Shut down the Integration Server Administrator. See the *webMethods Integration Server Administrator's Guide* for your release for the procedure to do this.
2. Edit the following file:

Integration Server_directory \config\redir.cnf

3. Add the following line to the file:

```
<value name="wm.art">false</value>
```

4. Save the file and restart Integration Server.

Clustering Considerations and Requirements

Note:

The following sections assume that you have already configured the Integration Server cluster. For details about webMethods clustering, see the *webMethods Integration Server Clustering Guide* for your release.

The following considerations and requirements apply to Lotus Notes Adapter in a clustered environment.

Requirements for Each Integration Server in a Cluster

The following table describes the requirements of each Integration Server in a given cluster:

All Integration Servers in a given cluster For Example... must have identical...

Integration Server versions	All Integration Servers in the cluster must be the same version, with the same service packs and fixes applied.
Adapter packages	All adapter packages on one Integration Server should be replicated to all other Integration Servers in the cluster.
Adapter versions	All adapters in the cluster must be the same version.
Adapter connections	If you configure a connection to the Domino Server, this connection must appear on all servers in the cluster so that any Integration Server in the cluster can handle a given request identically.
Adapter services	<p>If you configure a specific adapter service, this same adapter service must appear on all servers in the cluster so that any Integration Server in the cluster can handle the request identically.</p> <p>If you allow different Integration Servers to contain different services, you might not derive the full benefits of clustering. For example, if a client requests a service that resides on only one server, and that server is unavailable, the request cannot be successfully redirected to another server.</p>

For information about replicating adapter packages, connections, and adapter services across multiple Integration Servers in a cluster, see [“Replicating Packages to webMethods Integration Servers” on page 32](#).

Considerations When Installing Lotus Notes Adapter Packages

For each Integration Server in the cluster, use the standard Lotus Notes Adapter installation procedures for each machine, as described in [“Installing webMethods Lotus Notes Adapter 6.0” on page 21](#).

4 Adapter Connections

■ Overview	36
■ Before Configuring or Managing Adapter Connections	36
■ Configuring Lotus Notes Adapter Connections	36
■ Dynamically Changing a Service's Connection at Run Time	39
■ Viewing Adapter Connection Parameters	40
■ Editing Adapter Connections	41
■ Copying Adapter Connections	42
■ Deleting Adapter Connections	42
■ Enabling Adapter Connections	43
■ Disabling Adapter Connections	43

Overview

This chapter describes how to configure and manage Lotus Notes Adapter connections. For more information about how adapter connections work, see [“Adapter Connections” on page 13](#).

Before Configuring or Managing Adapter Connections

➤ To prepare to configure or manage adapter connections

1. Make sure you have Integration Server administrator privileges so that you can access the Lotus Notes Adapter's administrative screens. For information about setting user privileges, see the *webMethods Integration Server Administrator's Guide* for your release.
2. Set up your adapter's environment to work with the Lotus Domino classes and methods on your Lotus Notes system. For instructions, see [“Installing webMethods Lotus Notes Adapter 6.0” on page 21](#).
3. Start Integration Server and Integration Server Administrator if they are not already running.
4. Using the Integration Server Administrator, make sure the WmLotusNotesAdapter package is enabled. See [“Enabling Adapter Connections” on page 43](#) for instructions.
5. Using Designer, create a user-defined package for each connection you plan to configure. See [“Lotus Notes Adapter Package Management” on page 28](#) for more information about managing packages for the adapter.

Configuring Lotus Notes Adapter Connections

When you configure Lotus Notes Adapter connections, you specify information that Integration Server uses to connect to the Domino Server.

You create Lotus Notes Adapter connections using the Integration Server Administrator.

➤ To configure a connection

1. In the **Adapters** menu in the Integration Server Administrator navigation area, click **Lotus Notes Adapter**.
2. On the Connections screen, click **Configure New Connection**.
3. On the Connection Types screen, click **Lotus Notes Adapter Connection** to display the Configure Connection Type screen.
4. In the **Lotus Notes Adapter** section, provide values for the following fields.

Field	Description/Action
Package	The package in which to configure the connection. You must create the package using Designer before you can specify it using this parameter. For general information about creating packages, see the <i>webMethods Service Development Help</i> for your release. Note: Configure the connection in a user-defined package rather than in the adapter's package. See " Lotus Notes Adapter Package Management " on page 28 for other important considerations when creating packages for the Lotus Notes Adapter.
Folder Name	The folder in which to configure the connection.
Connection Name	The name you want to give the connection. Connection names cannot have spaces or use special characters reserved by Designer. For more information about the use of special characters in package, folder, and element names, see the <i>webMethods Service Development Help</i> for your release.

5. In the **Connection Properties** section, provide values for the fields noted below.

The values you enter in the **Connection Properties** section are not validated until run time when you execute the `createSession` method.

Field	Description/Action
Lotus Domino Host Name	Required. The host name or IP address of the machine on which your Domino Server is running. Optionally you can include the port number of the host. For example, any of the following values are valid: <ul style="list-style-type: none"> ■ my.server.com ■ my.server.com:81 ■ 127.0.0.1 ■ 127.0.0.1:81
Lotus Domino User name	The user name that you use to log into the Domino Server. If your Domino Server is configured to allow anonymous logins, you may leave this field blank.

Field	Description/Action
Lotus Domino password	The password that you use to log into the Domino Server. If your Domino Server is configured to allow anonymous logins, you may leave this field blank.
Retype Lotus Domino password	Re-type the password.
HTTP/HTTPS Connection	Select the type of connection that you use to connect to the Domino Server. For HTTPS connections, you must also copy the TrustedCerts.class file to the <i>Integration Server_directory</i> \packages\WmLotusNotesAdapter\code\classes directory. For more information about this file, see your Domino Server documentation. Ensure that your Domino server is correctly configured to accept SSL connections. See your Domino Server documentation.

6. In the **Connection Management Properties** section, Software AG recommends that you set the **Enable Connection Pooling** parameter to **false** and take the default value for the remaining fields. This is because there is no connection pooling support in Lotus Notes Adapter.

The fields in this section are described as follows:

Parameter	Description/Action
Enable Connection Pooling	Enables the adapter to use connection pooling. Default: true . If set to false , Lotus Notes Adapter does not use the connection pooling feature.
Minimum Pool Size	The minimum number of connection objects that remain in the connection pool at all times. When the adapter creates the pool, it creates this number of connections. Default: 1.
Maximum Pool Size	The maximum number of connection objects that can exist in the connection pool. When the connection pool has reached its maximum number of connections, the adapter will reuse any inactive connections in the pool or, if all connections are active, it will wait for a connection to become available. Default: 10.
Pool Increment Size	If connection pooling is enabled, this field specifies the number of connections by which the pool will be incremented if connections are

Parameter	Description/Action
	needed, up to the maximum pool size. Default: 1.
Block Timeout	If connection pooling is enabled, this field specifies the number of milliseconds that Integration Server will wait to obtain a connection with the Domino Server before it times out and returns an error. Default: 1000.
Expire Timeout	If connection pooling is enabled, this field specifies the number of milliseconds that an inactive connection can remain in the pool before it is closed and removed from the pool. For example, to specify 10 seconds, specify 10000. Enter 0 to specify no timeout. Default: 1000.
	<p>Note: The adapter will never violate the Minimum Connections parameter. These connections remain in the pool regardless of how long they are inactive.</p>
Startup Retry Count	If connection pooling is enabled, this field specifies the number of times that the system should attempt to initialize the connection pool at startup if the initial attempt fails, before issuing an AdapterConnectionException.
Startup Backoff Timeout	If connection pooling is enabled, this field specifies the number of seconds to wait between each attempt to initialize the connection pool.

7. Click **Save Connection**.

The connection you created appears on the adapter's Connections screen and in Designer.

Dynamically Changing a Service's Connection at Run Time

You may run a service using a connection other than the default connection that was associated with the service when the service was created. To override the default, you must code your flow to pass a value through the pipeline into a service's \$connectionName field.

For example, you have a flow whose primary purpose is to update your company's North American Lotus Domino Server. However, you want the flow to have the capability to update your company's European Lotus Domino Server-with the decision of which server to update to be made programmatically at run time. The output signature of the flow's first service contains a field called

Target. The flow could branch based on the value in Target. If Target contains the value "NA", the second service in the flow would ignore \$connectionName-thus using its default connection to connect to (and then update) the North American server. However, if Target contains the value "E", the second service in the flow would use the value in the \$connectionName from the pipeline and connect to (and then update) the European server.

Keep in mind these restrictions when using dynamic connections:

- Both connections - the default and override - must have the same Lotus Domino classes and methods.
- The \$connectionName field is present only in services created with Designer.

For more information, see [“Changing the Connection Associated with an Adapter Service at Run Time” on page 14.](#)

Viewing Adapter Connection Parameters

You can view a connection's parameters from the Integration Server Administrator, or Designer.

Viewing Adapter Connection Parameters from Integration Server Administrator

➤ To view the parameters for a connection using the Integration Server Administrator

1. In the **Adapters** menu in the navigation area of the Integration Server Administrator, click **Lotus Notes Adapter**.

When using the adapter with Integration Server 8.0 and later, you can sort and filter the list of connections that appears on the Connections screen.

- To sort information on the Connections screen, click the **Up** and **Down** arrows at the top of the column you want to sort.
- To filter the list of connections:
 1. On the Connections screen, click **Filter Connections**.
 2. Type the criterion by which you want to filter into the **Filter criteria** box. Filtering is based on the node name, not the connection alias. To locate all connections containing specific alphanumeric characters, use asterisks (*) as wildcards. For example, if you want to display all connections containing the string "abc", type *abc* in the **Filter criteria** box.
 3. Click **Submit**. The Connections screen displays the connections that match the filter criteria.
 4. To re-display all connections, click **Show All Connections**.

The Connections screen appears, listing all the current connections. You can control the number of connections that are displayed on this screen. For more information, see [“Controlling Pagination” on page 19](#).

2. On the Connections screen, click the  icon for the connection you want to see.

The View Connection screen displays the parameters for the connection. For descriptions of the connection parameters, see the table of parameters in [“Configuring Lotus Notes Adapter Connections” on page 36](#).

3. Click **Return to Lotus Notes Adapter Connections** to return to the main connections screen.

Viewing Adapter Connection Parameters from Designer

➤ To view the parameters for a connection using Designer

1. From the Designer navigation area, open the package and folder in which the connection is located.
2. Double-click the connection you want to view.

The parameters for the connection appear in the Connection Information tab. For descriptions of the connection parameters, see the table of parameters in [“Configuring Lotus Notes Adapter Connections” on page 36](#).

Editing Adapter Connections

If you want to redefine parameters that a connection uses when connecting to a Domino Server, you can update a connection's parameters using the Integration Server Administrator.

Note:

You must disable a connection before you can edit it. See [“Disabling Adapter Connections” on page 43](#) for details.

➤ To edit a connection

1. In the **Adapters** menu in the navigation area of the Integration Server Administrator, click **Lotus Notes Adapter**.
2. On the Connections screen, click the  icon for the connection you want to edit.

The Edit Connection screen displays the current parameters for the connection. Update the connection's parameters by typing or selecting the values you want to specify.

For descriptions of the connection parameters, see the table of parameters in [“Configuring Lotus Notes Adapter Connections” on page 36](#).

3. Click **Save Changes** to save the connection and return to the Connections screen.

Copying Adapter Connections

You copy Lotus Notes Adapter connections using the Integration Server Administrator. You can copy an existing Lotus Notes Adapter connection to create a new connection with the same or similar connection properties without the need to type all properties for the new connection.

› To copy a connection

1. In the **Adapters** menu in the navigation area of the Integration Server Administrator, click **Lotus Notes Adapter**.
2. On the Connections screen, click the  icon for the connection you want to copy.

The Copy Connection screen displays the current parameters for the connection you want to copy. Name the new connection and edit any connection parameters as needed by typing or selecting the values you want to specify.

For descriptions of the connection parameters, see the table of parameters in [“Configuring Lotus Notes Adapter Connections” on page 36](#).

3. Click **Save Connection** to save the connection and return to the Connections screen.

Note:

You must enter and then retype a password before you can save the new connection.

Deleting Adapter Connections

You delete connections using the Integration Server Administrator. If you no longer want to use a Lotus Notes Adapter connection, use the following instructions for deleting the connection.

Note:

You can delete a connection only if it is disabled, as described in [“Disabling Adapter Connections” on page 43](#).

Important:

If you delete a Lotus Notes Adapter connection, any adapter services that are defined to use the connection will no longer work. You cannot change the connection for an adapter service after it has been defined. You can change the parameters for the existing connection. See [“Editing Adapter Connections” on page 41](#) for instructions.

› To delete a connection

1. In the **Adapters** menu in the navigation area of the Integration Server Administrator, click **Lotus Notes Adapter**.

2. On the Connections screen, make sure that the connection is disabled. See “[Disabling Adapter Connections](#)” on page 43 for instructions.
3. On the Connections screen, click **X** for the connection you want to delete.

The Integration Server deletes the adapter connection.

Enabling Adapter Connections

You enable connections using the Integration Server Administrator. You must enable a Lotus Notes Adapter connection before you can configure an adapter service using the connection, or before an adapter service can use the connection at run time.

Note:

When you reload a package that contains enabled connections, the connections will automatically be enabled when the package reloads. If the package contains connections that are disabled, they will remain disabled when the package reloads.

> To enable a connection

1. In the **Adapters** menu in the navigation area of the Integration Server Administrator, click **Lotus Notes Adapter**.
2. On the Connections screen, click **No** in the **Enabled** column for the connection you want to enable.

The Integration Server Administrator enables the adapter connection and displays **✓** and **Yes** in the **Enabled** column.

Disabling Adapter Connections

You disable connections using the Integration Server Administrator. A Lotus Notes Adapter connection must be disabled before you can edit or delete it.

> To disable a connection

1. In the **Adapters** menu in the navigation area of the Integration Server Administrator, click **Lotus Notes Adapter**.
2. On the Connections screen, click **Yes** in the **Enabled** column for the connection you want to disable.

The adapter connection becomes disabled and you will no longer see a **No** in the **Enabled** column.

5 Adapter Services

■ Overview	46
■ Before Configuring or Managing Adapter Services	46
■ Using Lotus Notes Adapter Services	46
■ Configuring Lotus Domino Operation Services	49
■ Testing Adapter Services	50
■ Viewing Adapter Services	51
■ Editing Adapter Services	52
■ Deleting Adapter Services	52
■ Enabling Automatic Data Validation	53
■ Reloading Adapter Values	54
■ Reloading for a Single Adapter Service	54
■ Reloading for All Adapter Services	54
■ Editing the XML Data File	55

Overview

The following sections describe how to configure adapter services, which you use for webMethods-to-Lotus Notes Adapter communications.

Before you configure the Lotus Notes Adapter services, you must configure the connections you plan to use with them. See [“Configuring Lotus Notes Adapter Connections” on page 36](#) for details.

For a description of adapter services, see [“Adapter Services” on page 14](#).

Before Configuring or Managing Adapter Services

➤ To prepare to configure or manage Lotus Notes Adapter services

1. Start your webMethods Integration Server and the Integration Server Administrator if they are not already running.
2. Make sure you have Integration Server Administrator privileges so that you can access the Lotus Notes Adapter's administrative screens. For information about setting user privileges, see the *webMethods Integration Server Administrator's Guide* for your release.
3. Make sure that the XML data file, located in the *Integration Server_directory* \packages\WmLotusNotesAdapter\config directory, contains the definitions of the Lotus Domino classes and methods required for development. See [“Editing the XML Data File” on page 55](#) for instructions.
4. Using the Integration Server Administrator, make sure the WmLotusNotesAdapter package is enabled. See [“Enabling Packages” on page 29](#) for instructions.
5. Using the Integration Server Administrator, configure an adapter connection to use with the adapter service. See [“Enabling Adapter Connections” on page 43](#) for instructions.
6. Start Designer if it is not already running.

Note:

If you are using Designer, use the Service Development perspective. For more information, see the *webMethods Service Development Help* for your release.

7. Using Designer, create a user-defined package to contain the service, if you have not already done so. When you configure adapter services, you should always define them in user-defined packages rather than in the WmLotusNotesAdapter package. For more information about managing packages for the adapter, see [“Package Management” on page 27](#).

Using Lotus Notes Adapter Services

The following table lists the tasks required to use adapter services:

Task	Use this tool...
1 Configure an adapter connection. See “Configuring Lotus Notes Adapter Connections” on page 36 for details.	Integration Server (Adapters menu)
2 Configure the adapter service using the adapter service template. You specify the: <ul style="list-style-type: none"> ■ Adapter connection ■ Lotus Domino Class ■ Domino Class Method ■ Primitiveoption See “Configuring Lotus Domino Operation Services” on page 49 for more information about configuring adapter services.	Designer
3 If you plan to use an Integration Server flow or Java service to invoke the adapter service, design the flow or Java service to use this adapter service. For details, see the <i>webMethods Service Development Help</i> for your release.	Designer
<p>Important:All services must first invoke the adapter service that specifies the <code>lotus.domino.Session</code> class and <code>createSession</code> method to create the session object for the Domino Server. For more information about how to configure this service, see “Example Lotus Notes Operation” on page 47. Additionally, the flow or Java service must call the <code>recycle</code> and/or <code>logout</code> service, as appropriate. For more information about these services, see “Freeing Resources After Processing” on page 16.</p>	
4 Manage the adapter service. See “Lotus Notes Adapter Package Management” on page 28 and “Built-In Services Reference” on page 77 for details.	Designer and Integration Server

Example Lotus Notes Operation

This example shows the steps necessary to use the Lotus Notes Adapter to configure an adapter connection and the necessary adapter services that, when called from a flow service, create a database on the Domino Server:

1. In Designer, create a new package, for example, LotusTest.
2. In the Integration Server Administrator, configure a new Lotus Notes connection in the package you created, for example, ConnectionTest. Be sure to enable the connection.
3. In Designer, create a folder for new adapter services, for example, Services.
4. In Designer, configure the following new adapter services using the Domino Operation template and the connection you created in step 2:
 - a. An adapter service that creates the session object on the Domino Server, for example, createSessionService. In the adapter service template, select **lotus.domino.session** as the class, and **createSession** as the method.
 - b. An adapter service that creates a getDbDirectory object on the Domino Server, for example, getDbDirObject. In the adapter service template, select **lotus.domino.session** as the class, and **getDbDirectory** as the method.
 - c. An adapter service that creates a new, blank database, for example, createDB. In the adapter service template, select the **lotus.domino.DbDirectory** class, and the **createDatabase** method.
5. Create a new empty flow service (for example, CreateDbFlow) and call the adapter services you created in step 4 in the following sequence:

createSessionService

getDbDirObject

createDB

6. As the last step in flow service, call the `wm.ln.authentication:logout` adapter service. For information about this adapter service, see [“Freeing Resources After Processing” on page 16](#).
7. In the flow service, map the fields in the pipeline.

Using the example adapter services created in step 4, the mapping is as follows:

- **createSessionService**. Inputs are user and password (default reads from the connection or a different user and password can be provided). Output is session.
 - **getDbDirObject**. Inputs are server and session (leave the server field blank and map the session from the CreateSessionService service). Output is dbDirectory.
 - **createDB**. Inputs are dbDirectory and dbfile (map the dbDirectory field from the getDbDirObject service and provide the name you want for the database in the dbfile field). Output is database.
8. Run the CreateDbFlow flow service to create a database on the Domino Server with the name specified in the dbfile field.

Configuring Lotus Domino Operation Services

When you configure adapter services, define them in user-defined packages rather than in the WmLotusNotesAdapter package. This helps you manage the package more easily. You create this package before you configure the adapter service. For details, see [“Lotus Notes Adapter Package Management” on page 28](#).

The Lotus Notes Adapter installation includes a single template from which all Lotus Notes services can be configured through Designer.

Note:

Before you configure an adapter service, you must create the adapter connection you plan to use with it. See [“Adapter Connections” on page 35](#).

➤ To configure an adapter service

1. Review the steps in [“Before Configuring or Managing Adapter Services” on page 46](#).
2. Start Designer.
3. If you are using Designer, perform the following:
 - a. Right-click the package in which the service should be contained and select **New > Adapter Service**.
 - b. Select the parent namespace and type a name for the adapter service.
 - c. Click **Next**.
4. Select **Lotus Notes Adapter** as the adapter type and click **Next**.
5. Select the appropriate **Adapter Connection Name** and click **Next**.
6. Select the **Domino Operation** template and if you are using Designer, click **Finish**. The service is created and its parameters and controls are displayed in the Designer.
7. Select the **Domino Operation** tab and specify the following values:

Field	Description/Action
Lotus Domino Class	<p>Displays the available classes. Select the Domino class that provides the type of action you want to perform:</p> <ul style="list-style-type: none"> ■ If you are configuring an adapter service that creates the session object for the Domino Server, select the lotus.domino.Session class.

Field	Description/Action
	<ul style="list-style-type: none"> ■ If you are configuring an adapter service for any other Domino operation, select the class for that operation. See the Lotus Domino Toolkit for Java/CORBA for information about the classes.
Domino Class Method	<p>Displays the methods for the selected class. Select the appropriate method that is available on the selected Domino object:</p> <ul style="list-style-type: none"> ■ If you are configuring an adapter service that creates the session object for the Domino Server, select the createSession method. ■ If you are configuring an adapter service for any other Domino operation, select the method for that operation. See the Lotus Domino Toolkit for Java/CORBA for information about a method.
Primitive	<p>By default the adapter service generates the input and output signatures as Java data primitives. Do not select this option to use the default.</p> <p>However, you can force the adapter service to generate the signatures as Strings. In this case, select this option.</p>

Note:

See “[Built-In Services Reference](#)” on page 77 for information about the adapter services and how they are provided in Designer.

- The input and output signature parameters for the selected class and method are shown below the **Primitive** check box. For information about the parameters and return values for the selected class and method, see the *Domino Objects for Java Reference Guide* in the Lotus Domino Toolkit for Java/CORBA documentation.
- You can select the **Adapter Settings** tab at any time to confirm adapter properties such as adapter type, connection name, and service template, as needed.
- For information about using the **Input/Output** and **Settings** tabs, see the *webMethods Service Development Help* for your release.
- From the **File** menu, select **Save**.

Testing Adapter Services

You use Designer to test adapter services.

For more information about testing and debugging services, see the *webMethods Service Development Help* for your release.

When you enter the values for the initial connection to the Domino Server, use your Lotus Notes user ID and password.

➤ To test adapter services

1. Review the steps in [“Before Configuring or Managing Adapter Services”](#) on page 46.
2. In Designer, expand the package and folder that contain the service you want to test.
3. Create a new flow service.
4. Insert a flow step that calls the adapter service that invokes the `createSession` method in the `lotus.domino.Session` class.
5. Insert the flow steps necessary to call the adapter service that performs the Domino operation you want to test. Sometimes you need to call more than one service before you can obtain the necessary object. See the reference information in [“Built-In Services Reference”](#) on page 88, and also consult the Lotus Domino Toolkit for Java/CORBA documentation as necessary to become familiar with the services.

Note:

A samples package is available that contains examples and instructions for using certain methods. You can browse the samples package and see if there is a sample that fits the type of operation you want to test. For more information about obtaining the sample package, see [“Adapter Services”](#) on page 14.

6. If you are using Designer, select **Run > Run As > Run Service**.
7. For every service input field, you will be prompted to enter an input value. Enter a value for each input field and then click **OK**.
8. Click the **Service Result** tab (in Designer) to view the output from this service.

Viewing Adapter Services

You use Designer to view adapter services.

➤ To view a service

1. Review the steps in [“Before Configuring or Managing Adapter Services”](#) on page 46.
2. In Designer, expand the package and folder that contain the service you want to view.
3. Double-click the service you want to view.

Designer displays the configured service in the service template's Adapter Service Editor.

Editing Adapter Services

You use Designer to edit adapter services.

> To edit an adapter service

1. In Designer, browse to and open the adapter service that you want to edit.
2. Double-click the service that you want to edit.

Designer displays the adapter service in the service template's Adapter Service Editor.

3. Do one of the following:
 - If you have the VCS Integration feature enabled, right-click the service and select **Check Out**.
 - If you do not have the VCS Integration feature enabled, right-click the service and select **Lock for Edit**.
 - If you are using the local service development feature, from the **Team** menu in Designer, select the appropriate option to check out the service. The options available in the **Team** menu depend on the VCS client that you use.
4. Modify the values for the adapter service's parameters as needed. For detailed descriptions of the service's parameters, see the section on configuring a service for the specific type of service you want to edit.
5. After you complete your modifications, save the service and do one of the following:
 - If you have the VCS Integration feature enabled, right-click the service and select **Check In**. Enter a check-in comment and click **OK**.
 - If you do not have the VCS Integration feature enabled, right-click the service and select **Unlock**.
 - If you are using the local service development feature, from the **Team** menu in Designer, select the appropriate option to check in the service. The options available in the **Team** menu depend on the VCS client that you use.
6. Save the service.

Deleting Adapter Services

You use Designer to delete adapter services.

> To delete an adapter service

1. Review the steps in [“Before Configuring or Managing Adapter Services”](#) on page 46.
2. In Designer, expand the package and folder that contain the service you want to delete.
3. Right-click the adapter service and click **Delete**.

Enabling Automatic Data Validation

Designer enables the Lotus Notes Adapter to validate user-defined data for adapter services at design time. You can validate the values for a single adapter service or you can configure Designer to always validate the values for adapter services. Both options could potentially slow your design-time operations.

When you enable data validation for a single adapter service, Designer compares the service values against the resource data that has already been fetched from the selected adapter.

If you select the option to always validate values for adapter services, it will do so for all webMethods WmART-based adapters installed on Integration Server.

For more information about the **Adapter Service/Notification Editor** and other Designer menu options and toolbar icons, see the *webMethods Service Development Help* for your release.

Validation for a Single Adapter Service

➤ To enable automatic data validation for a single adapter service

1. Review the steps in [“Before Configuring or Managing Adapter Services”](#) on page 46.
2. In Designer, expand the package and folder that contain the service for which you want to enable automatic validation.
3. Double-click the service for which you want to validate the data.

Designer displays the configured adapter service in the service template's Adapter Service Editor.

4. Click the  icon.

Validation for All Adapter Services

➤ To always validate the values for all adapter services

1. Review the steps in [“Before Configuring or Managing Adapter Services”](#) on page 46.
2. Start Designer.

3. If you are using Designer, select the **Window > Preferences > Software AG > Service Development > Adapter Service/Notification Editor** item.
4. Enable the **Automatic data validation** option.
5. Click **OK**.

Reloading Adapter Values

You can enable the Lotus Notes Adapter to reload and validate user-defined data for adapter services at design time in Designer. You can reload values for a single adapter service or you can configure Designer so it automatically reloads the values for adapter services. Both options could potentially slow your design-time operations.

When you reload adapter values for a single adapter service, Designer compares the service values against the resource data that has already been fetched from the selected adapter.

If you select the option to always reload values for adapter services, it will do so for all webMethods WmART-based adapters installed on Integration Server.

For more information about the **Adapter Service/Notification Editor**, other menu options, and toolbar icons, see the *webMethods Service Development Help* for your release.

Reloading for a Single Adapter Service

➤ To reload the adapter values for a single adapter service

1. Review the steps in “[Before Configuring or Managing Adapter Services](#)” on page 46.
2. In Designer, expand the package and folder that contain the service for which you want to enable automatic validation.
3. Double-click the service for which you want to validate the data.

Designer displays the configured adapter service in the service template's Adapter Service Editor.

4. Click the  icon.

Note: This icon is available only when the **Automatic polling of adapter metadata** option is disabled.

Reloading for All Adapter Services

➤ To reload the adapter values for a single adapter service

1. Review the steps in [“Before Configuring or Managing Adapter Services”](#) on page 46.
2. Start Designer.
3. If you are using Designer, select the **Window > Preferences > Software AG > Service Development > Adapter Service/Notification Editor** item.
4. Enable the **Automatic polling of adapter metadata** option.
5. Click **OK**.

Editing the XML Data File

When configuring an adapter service, you select a Lotus Domino class and method to call. These classes and methods are defined in an XML data file. You may edit the XML data file to remove classes so that the list is smaller, or to expose only some classes for security reasons. The `WmXMLDataFile.xml` file can be found in the *Integration Server_directory* \packages\WmLotusNotesAdapter\config directory. For more information about adapter services, see [“Adapter Services”](#) on page 14.

Note:

Be sure to use an XML editor when making changes to this file. Any changes made must result in a valid XML file. If the XML data file is invalid, the adapter may fail during the following operations: while loading the XML file, while configuring an adapter service, or while loading or executing a service.

➤ To edit the XML data file

1. Navigate to the *Integration Server_directory* \packages\WmLotusNotesAdapter\config directory.
2. Open the `WmXMLDataFile.xml` file with an XML editor.
3. Each class is located within the `<Class name>` tag. Each method for the particular class is located within the `<method>` tag.
 - To delete a class, delete all text from the applicable `<Class name="lotus.domino.xx.XXX" instance="acl">` tag to the corresponding `</Class>` tag.
 - To delete a method, delete all text from the applicable `<method>` tag to the corresponding `</method>` tag.
4. Save the file.
5. Reload the `WmLotusNotesAdapter` package for the changes to take effect. For more information about loading packages, see [“Loading, Reloading, and Unloading Packages”](#) on page 30.

6 Lotus Notes-to-webMethods Communications

■ Overview	58
■ Setting Up the Domino Server for Outbound Transactions	58
■ Using SSL to Communicate with Integration Server	61

Overview

This chapter provides instructions for creating Lotus Notes-to-webMethods (outbound) transaction services. To create Lotus Notes-to-webMethods services, you set up your Lotus Notes Designer to implement shared code. You also set up your Integration Server to handle the messages as they arrive from the Domino Server.

This chapter provides sample code to implement your outbound transactions. Additionally, it provides information about how to configure your Lotus Notes system to communicate with the Integration Server using SSL and describes how to monitor your outbound Lotus Notes request messages at run time.

For a detailed description of Lotus Notes-to-webMethods communications, see [“Lotus Notes-to-webMethods Communications” on page 17](#).

Setting Up the Domino Server for Outbound Transactions

To set up the Domino Server to enable outbound communications to webMethods, perform the following steps:

Step	See
1	“Set the Classpath in the Lotus Notes notes.ini File” on page 58
2	“Write Shared Agent Code” on page 59

Set the Classpath in the Lotus Notes notes.ini File

➤ **To set the Classpath in the Lotus Notes notes.ini file to compile shared code**

1. From the Domino Server directory, open the notes.ini file.

Note:

The Classpath should be set on the server and not on the client.

2. Add the following line in the notes.ini file:

```
JavaUserClasses=  
    Software AG_directory  
    \common\lib\wm-isclient.jar; LotusNotesDesigner_directory\Notes.jar
```

Replace *Integration Server_directory* and *LotusNotesDesigner_directory* with the appropriate directory in which the products are installed.

If the client.jar file is not available locally, copy the file to a local directory and add that directory to the Classpath.

Write Shared Agent Code

Lotus Notes provides the option of using Java or LotusScript to programmatically manipulate data within the database. The Java or LotusScript code is stored in modules called Agents. Agents can be called manually or through some automated method such as a trigger in response to a specific database event, or at a scheduled time.

For information about building a Java client, see the *webMethods Service Development Help* for your release. For information about working with agents, see the Lotus Domino Toolkit for Java/CORBA documentation.

Complete the following instructions to create the agent code in Java or LotusScript.

Using Sample Shared Agent Code in Java

➤ To implement the sample shared agent code in Java

1. Open the Lotus Notes Designer.
2. Open a database.
3. Click **Shared Code**, then click **Agents**.
4. Click the **New Agents** tab and enter a name for the new agent, then close the dialog box.
5. In the Agent window, select **Java** from the drop-down menu.
6. Copy and paste the following sample code.

```
import lotus.domino.*;//STANDARD DOMINO CLASSES
import java.io.*;//INPUT OUTPUT CLASSES
import java.util.*;//UTILITIES PACKAGES
import com.wm.app.b2b.client.*;
import com.wm.data.*;

public class JavaAgent extends AgentBase {
public void NotesMain() {
try {
//SETUP MAIN OBJECTS
Session oSession = getSession();
AgentContext oAgentContext = oSession.getAgentContext();
Database oDatabase = oAgentContext.getCurrentDatabase();

//CALLING VendorDB:getVendorList
Context c= new Context();
c.connect("localhost:5555", "Administrator", "manage");
IData results = c.invoke("VendorDB", "getVendorList", null);

//CREATE AGENT OUTPUT CLASS
PrintWriter oPrint = getAgentOutput();
```

```

//GET WEB DOCUMENT
Document oWebDocument = oAgentContext.getDocumentContext();
//SETUP DATE OBJECT FOR TIME STAMP
    DateTime oDate = oSession.createDateTime("Today");

oDate.setNow();
//VARIABLES
Double vLoop = new Double(0);
String vHTML = new String("");
String vUNID = new String("");
int vLoop1= 5

//***** PROCESS SELECTED DOCUMENTS *****

    vHTML = vHTML + "<FONT Face=verdana SIZE=4><B>List of
Vendors</B></FONT><BR><BR>";
    vHTML = vHTML + "<TABLE STYLE=\"font:8ptVerdana\">";
    vHTML = vHTML +
"<TR><TD><B>Name</B></TD>
<TD><B>Company</B></TD>
<TD><B>Status</B></TD></TR>";

        if (results!=null)
        {
            IDataCursor _cursor = results.getCursor();
            IData[] _vendor= IDataUtil.getIDataArray(_cursor, "Vendors");
            IDataCursor _vendCursor= null
            if (_vendor.length >0)
            {
                for (int i=0; i < _vendor.length; ++i)
                {
                    _vendCursor = _vendor[i].getCursor();
                    vUNID=IDataUtil.getString(_vendCursor, "Id");
                    vHTML = vHTML + "<TR><TD>";
                    vHTML = vHTML + vUNID + "</TD><TD >

<a href=./VendorDetails?OpenForm&Name="+IDataUtil.getString(_vendCursor,
"Name")+"">" + IDataUtil.getString(_vendCursor, "Name")+ "</a></TD><TD>";
                    vHTML = vHTML +IDataUtil.getString(_vendCursor, "Company")+
"<TD><FONTCOLOR=\"red\">"+IDataUtil.getString(_vendCursor,
"Status")+ "</FONT><BR>";
                    vHTML = vHTML + "</TD></TR>";
                } //end-for
            } //end-if(_vendor)
            _vendCursor.destroy();
            _cursor.destroy();
        } //end-if

// PROCESS DOCUMENT
vHTML = vHTML + "</TABLE>";
oPrint.println(vHTML);
    } catch(Exception e)
    {
        e.printStackTrace();
    }
}
}

```

7. Save your changes and the code will compile.
8. Work with your Lotus Domino application developer to ensure that the shared agent is executed according to your integration needs. You may also need to set up the Domino Server to put in place the mechanisms to initiate the events that the shared agent will use to start the service.

Using Sample Shared Agent Code in LotusScript

If you would like to implement Lotus Notes-to-webMethods communications in C/C++, Integration Server provides the `webMethods.dll` for the Windows platform. For the UNIX platform, `webMethods` provides the corresponding `so/sl` files.

Starting with Integration Server 7.1, the pre-built C library files are no longer included in Integration Server, or Designer. You must build the C library files for your system using the SDK. For information about building a C or C++ client, see the *webMethods Integration Server C/C++ API Reference* or *webMethods Service Development Help* for your release.

After you complete your development, you may execute it from your LotusScript.

Using SSL to Communicate with Integration Server

Before you set up your Lotus Notes system to use SSL to communicate with the Integration Server, you must first configure the Integration Server to enable an HTTPS port. If you need instructions for setting up an HTTPS port, see the *webMethods Integration Server Administrator's Guide* for your release.

For more information about the methods for setting the SSL option on the agent clients, see the `com.wm.app.b2b.client.BaseContext` class and the `com.wm.app.b2b.client.Context` class in the *webMethods Integration Server Java API Reference*.

7 Predefined Health Indicator

■ Predefined Health Indicator	64
-------------------------------------	----

Predefined Health Indicator

Microservices Runtime includes predefined health indicators for some of its basic components. The health indicator captures the connection details for all the WmART based adapters at runtime. For more information, see *webMethods Adapter Runtime User's Guide*.

8 Administrator APIs

■ Administrator APIs	66
----------------------------	----

Administrator APIs

The Administrator APIs are available for Lotus Notes Adapter. For more information about Administrator APIs and samples, see *webMethods Adapter Runtime User's Guide*.

9 Configuration Variables Templates for Adapter Assets in Microservices Runtime

- Configuration Variables Templates for Adapter Assets in Microservices Runtime 68

Configuration Variables Templates for Adapter Assets in Microservices Runtime

The webMethods Adapter Runtime (ART) asset properties that can be configured from Integration Server Administrator are available in the configuration variables template (`application.properties` file) generated by Microservices Runtime. For more information, see *webMethods Adapter Runtime User's Guide* and *Developing Microservices with webMethods Microservices Runtime*.

10 Logging and Exception Handling

■ Overview	70
■ Lotus Notes Adapter Logging Levels	70
■ Lotus Notes Adapter Message Logging	71
■ Handling Lotus Notes Exceptions	72
■ Lotus Notes Adapter Error Codes	72

Overview

The following sections describe message logging and exception handling for Lotus Notes Adapter. A list of error codes and supporting information appears in “[Lotus Notes Adapter Error Codes](#)” on [page 72](#).

Lotus Notes Adapter Logging Levels

Lotus Notes Adapter uses the Integration Server logging mechanism to log messages. You can configure and view the Integration Server logs to monitor and troubleshoot the Lotus Notes Adapter. For detailed information about logging in Integration Server, including instructions for configuring and viewing the different kinds of logs supported by the server, see the *webMethods Integration Server Administrator's Guide* for your release.

Configuring Adapter Logging Levels

Beginning with Integration Server 7.1, you can configure different logging levels for the Lotus Notes Adapter.

Accessing the Adapter's Logging Information

➤ **To access the adapter's logging information**

1. From the Integration Server Administrator screen, select **Settings > Logging**.

The **Logging Settings** screen appears. The **Loggers** section has **Adapters** included in the **Facility** section.

2. Expand the **Adapters** tree to see a list of all installed adapters with their code number and adapter description, along with the logging level.

Changing Logging Settings for the Adapter

➤ **To change logging settings for the adapter**

1. Click **Edit Logging Settings**. Select the required **Level of Logging** for the Lotus Notes Adapter.
2. After making your changes, click **Save Changes**.
3. For complete information about specifying the amount and type of information to include in the log, see the *webMethods Audit Logging Guide* for your release.

Lotus Notes Adapter Message Logging

Integration Server maintains several types of logs; however, Lotus Notes Adapter only logs messages to the audit, error and server logs. Because Lotus Notes Adapter works in conjunction with the WmART package, the adapter's messages and exceptions typically appear within log messages for the WmART package.

The logging levels are different depending on which version of Integration Server you are running the adapter on, as shown in the following table.

Integration Server	Log	Description
Integration Server 6.5	Audit Log	You can monitor individual adapter services using the audit log as you would audit any service in Integration Server. The audit properties for an adapter service are available in each Lotus Notes Adapter service template on the Audit tab.
	Error Log	Lotus Notes Adapter automatically posts critical-level and error-level log messages to the server's error log. These log messages will appear as adapter run-time messages.
	Server Log	Lotus Notes Adapter posts messages to the Server log, depending on how the server log is configured. Critical-level through debug-level log messages appear as adapter run-time log messages. V1-Verbose1 or V4-Verbose4 log messages appear as Lotus Notes Adapter log messages.
Integration Server 7.1 and higher	Audit Log	You can monitor individual adapter services using the audit log as you would audit any service in Integration Server. The audit properties for an adapter service are available in each Lotus Notes Adapter service template on the Audit tab.
	Error Log	Lotus Notes Adapter automatically posts fatal-level and error-level log messages to the server's error log. These log messages will appear as adapter run-time messages.
	Server Log	Lotus Notes Adapter posts messages to the server log, depending on how the server log is configured. Fatal-level through debug-level log messages appear as adapter run-time log messages. Trace-level log messages appear as Lotus Notes Adapter log messages.

The Lotus Notes Adapter's log messages appear in the format `ADA.0720.nnnnc`, where:

- `ADA` is the facility code that indicates that the message is from an adapter.
- `0720` (or `720`) is the Lotus Notes Adapter major code, which indicates that the error is generated by Lotus Notes Adapter.
- `nnnn` represents the error's minor code. For detailed descriptions of the Lotus Notes Adapter's minor codes, see “[Lotus Notes Adapter Error Codes](#)” on page 72.
- `c` represents the message's severity level (optional).

To monitor the Lotus Notes Adapter's log messages in the Server log, ensure that your server log's logging settings are configured to monitor the following facilities:

- 0113 Adapter Runtime (Managed Object)
- 0114 Adapter Runtime
- 0115 Adapter Runtime (Listener)
- 0117 Adapter Runtime (Adapter Service)
- 0118 Adapter Runtime (Connection)
- 0121 Adapter Runtime (SCC Transaction Manager)
- 0126 Adapter Runtime (SCC Connection Manager)

Handling Lotus Notes Exceptions

If a Lotus Notes Adapter object (for example, a connection or service) encounters an error with the Domino Server, it will throw an adapter error coupled with the Lotus Notes error, exactly as it was thrown by Lotus Notes. The Lotus Notes errors will be in an `IData` format.

For example, if a `getDbdirectory` service fails on the Domino Server at run time because its input fields contain invalid key values, you will receive an adapter error that indicates that the `getDbdirectory` service failed, and the adapter error will contain the specific error generated on the Domino Server indicating why the service failed. In this case, you would receive a Lotus Notes error specifying that the input key field values are invalid.

Lotus Notes Adapter Error Codes

This section lists the Lotus Notes Adapter's Critical- and Error-level minor codes, including the text of each error, information about the cause of the errors, and possible responses to the errors.

Code Description

106 Lotus Notes Adapter failed to load XMLDataFile.

Explanation: This message is thrown as a `ServiceException` wrapping the exception encountered while loading the XML data file.

Code Description

Action: The exception wrapped will have details on the nature of the failure.

107 Lotus Notes Adapter cannot load class: class-name.

Explanation: This message is encountered mainly during development time. This message indicates that the adapter is unable to load a Java class as specified in the XML data file.

Action: Verify that the classpath and the name of the class are correct.

511 Lotus Notes Adapter will not attempt to execute method: method-name **in class** class-name **using connection node:** connection_name **because no object reference was provided in the parameter-name parameter to execute such method on.**

Explanation: The Lotus Domino Toolkit for Java/CORBA requires an object in order to execute Lotus Notes Adapter. This error message indicates that an object was not provided.

Action: Check the flow or services to ensure that the object is properly mapped/provided.

513 Lotus Notes Adapter failed to execute method: method-name **in class** class-name. **Reason:** reason.

Explanation: This message is provided in the event that an exception other than a NotesException is thrown by a Domino object method invocation.

Action: Depending on the exception reported, make the necessary changes in input or on the Domino Server.

514 Lotus Notes Adapter failed to execute method method-name **in class** class-name. **Reason:** reason. **Exception ID:** id.

Explanation: Lotus Notes Adapter caught a NotesException. Lotus Notes Adapter extracts the ID of the exception and the text provided with it.

Action: Consult the appropriate IBM documentation, and make the necessary changes to work around the exception. This is an application-level error; however, it may also be a system-level error which requires a fresh session. The adapter may handle these errors automatically. For more information about exception IDs and the *Domino Objects for Java Reference Guide* in the Lotus Domino Toolkit for Java/CORBA documentation.

516 Lotus Notes Adapter failed to execute method: method-name **in class:** class-name, **because the indicated class can not be loaded.**

Explanation: This message is encountered mainly during execution time. This message indicates that the adapter is unable to load a Java class as specified in the XML data file. It may be seen if the classpath was altered after a service has been properly generated.

Action: Verify that the classpath and the name of the class are correct.

517 Lotus Notes Adapter failed to find the class-name **class definition in the XML data file. Aborting** service_name **execution.**

Code Description

Explanation: This exception is issued during run time. It indicates that a class has been removed from the XML data file after an adapter service has been created for it.

Action: Edit the XML data file to define the class and lists its methods. For instructions, see [“Editing the XML Data File” on page 55](#).

518 Lotus Notes Adapter failed to find the method-name method definition in class class-name.

Explanation: A method in a class that is defined in the XML data file does not exist.

Action: Verify that the definition of the method is correct in the XML data file. Also check the spelling of the method, and ensure that the signature is accurate.

519 Lotus Notes Adapter encountered an IllegalAccessException while executing the method-name method in class class-name.

Explanation: This exception indicates that the adapter does not have the proper permissions to execute certain Java code. This is a rare exception because the proper permissions should already be in place to execute Java methods in the Lotus Domino Toolkit for Java/CORBA.

Action: Ensure that the environment for the adapter is set up to allow the adapter to properly load and access the classes defined in the XML data file.

520 Lotus Notes Adapter will not attempt to execute method: method-name in class: class-name because it encountered a NumberFormatException while trying to convert input parameter: parm-name with value: value to type: type.

Explanation: This message occurs when the adapter cannot properly convert a parameter to the desired target. Adapter services that have a signature in primitive mode and one of the parameters is of a format that requires conversion (for example byte, int, long, float, double or short), may encounter this error.

Action: Correct the specified value and rerun the service.

521 Lotus Notes Adapter failed to execute the built-in service service-name. Reason: reason. Exception ID: id

Explanation: Lotus Notes Adapter caught a NotesException. Lotus Notes Adapter extracts the ID of the exception and the text provided with it.

Action: Consult the appropriate IBM documentation, and make the necessary changes to work around the exception. This is an application-level error; however, it may also be a system-level error which requires a fresh session. The adapter may handle these errors automatically. For more information about exception IDs and the associated constants that are defined in NotesError for the lotus.domino classes, see the *Domino Objects for Java Reference Guide* in the Lotus Domino Toolkit for Java/CORBA documentation.

522 Lotus Notes Adapter failed to execute the built-in service service-name. Reason: reason.

Explanation: This message is provided in the event that an exception other than a NotesException is thrown by a Domino object method invocation.

Code Description

Action: Depending on the exception reported, make the necessary changes in input or on the Domino Server.

523 Lotus Notes Adapter will not execute the `wm.ln.authentication:logout` service because there was no session object in the pipeline.

Explanation: The `wm.ln.authentication:logout` service requires that a session object be available in the pipeline under parameter name "session". If it is missing this message is encountered.

Action: Ensure that the session object is available in the pipeline.

600 Lotus Notes Adapter failed to initialize connection: connection_name to host: Lotus-Domino-server with user: user-name. Reason: reason. Possible reason: reason. Exception ID: id. Exception message: message-text.

Explanation: This error message appears if the adapter encounters an error when trying to establish a connection to a Domino Server for the specified user.

Action: Consult the appropriate IBM documentation, and make the necessary changes based on the exception reported. For more information about exception IDs and the associated constants that are defined in `NotesError` for the `lotus.domino` classes, see the *Domino Objects for Java Reference Guide* in the Lotus Domino Toolkit for Java/CORBA documentation.

A Built-In Services Reference

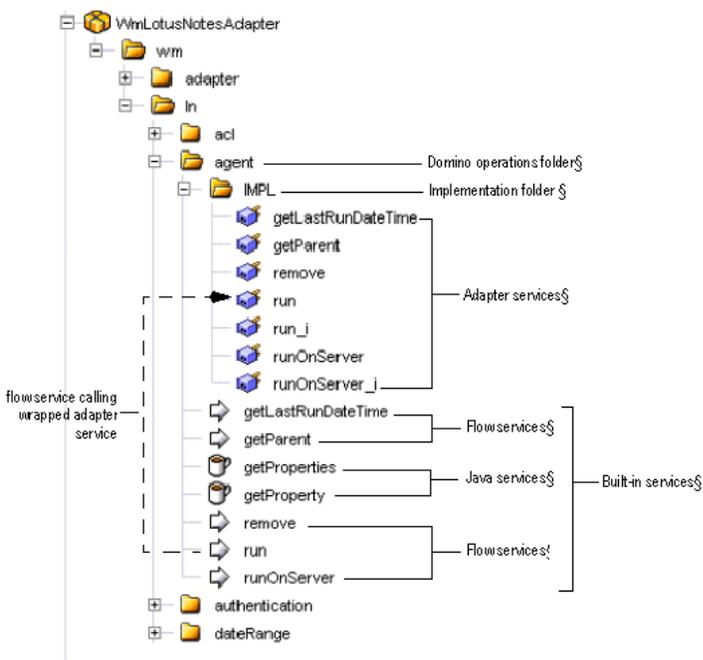
■ Overview	78
■ Domino Operations Folder Structure	78
■ Summary of Built-In Services	79
■ Built-In Services Reference	88

Overview

The `wm.In` folder includes subfolders that contain services for Domino operations, as well as additional folders that contain supporting utility services or records. This appendix explains the folder structure of the Domino operations folders, and provides reference information for the built-in flow and Java services.

Domino Operations Folder Structure

Some subfolders in the `wm.In` folder in the `WmLotusNotesAdapter` package each contain a set of flow services, Java services, adapter services, and utility services that support a group of Domino operations that are performed on the Domino Server. For example, the following diagram shows the contents of the `wm.In.agent` folder as it appears in Designer:



- **Domino operations folder.** Contains the built-in services and supporting adapter services for the Domino operation.
- **Built-in services.** Contains flow and Java services:
 - **Flow services.** Wraps one or more methods for a Java `lotus.domino` class. Integration developers use these services in their solutions. Within the flow, calls are made to the underlying adapter service in the `IMPL` folder. Input and output signatures in the flow service provide and acquire the parameters at the root level of the pipeline, simplifying the signatures specified in the underlying adapter service. Some flow services provide additional processing value while others serve only as a wrapper for a `lotus.domino.class` and method.

- **Java services.** Java programs that provide some type of conversion/utility and possibly wrap one or more methods in the Lotus Domino Toolkit for Java/CORBA.

For a listing of the built-in flow and Java services provided by the adapter, see [“Summary of Built-In Services” on page 79](#).

- **Implementation folder.** The IMPL folder contains the supporting adapter services called by the flow services:
 - **Adapter services.** Called by flow and Java services, the adapter services in the IMPL folder are preconfigured to wrap the Java Domino object methods for a Java lotus.domino class in the Lotus Domino Toolkit for Java/CORBA. They are designed to be used within a flow service, however, you can call them directly. For information about the inputs and outputs used by the predefined adapter services, see the documentation for the Lotus Domino Toolkit for Java/CORBA.

Summary of Built-In Services

Lotus Notes Adapter provides flow services for a subset of the Domino objects, and Java services for additional processing functionality. The table below lists the contents of each folder. However, because the flow services call adapter services that wrap the Domino objects, complete information about the object's classes and methods and their parameter definitions can be found in the Lotus Domino Toolkit for Java/CORBA documentation.

For input and output parameter information for these services, see [“Built-In Services Reference” on page 88](#).

Folder	Built-in Services
wm.ln.acl	The following flow services are included:
	aclEntryRecycle getEntry
	aclRecycle getFirstEntry
	addRole getNextEntry
	createACLEntry isRoleEnabled
	deleteRole remove
	disableRole removeACLEntry
	enableRole renameRole
	getACLEntryRoles save
	getACLRoles
	The following Java services are included:
	getACLEntryProperties getACLProperty

Folder	Built-in Services	
	getACLEntryProperty	setACLEntryProperty
	getACLProperties	setACLProperty
wm.In.agent	The following flow services are included:	
	getLastRunDateTime	run
	getParent	runOnServer
	remove	
	The following Java services are included:	
	getProperties	getProperty
wm.In.authentication	Provides services that assist in logging in and out of a session. The following flow services are included:	
	login	
	The following Java service is included:	
	logout	
wm.In.dateRange	The following flow services are included:	
	createDRFromString	getText
	getEndDateTime	setEndDateTime
	getStartDateTime	setStartDateTime
	The following Java service is included:	
	createDR	
wm.In.dateTime	The following flow services are included:	
	createDT	createDTFromString
	createDTFromDate	
	The following Java services are included:	
	createDT_old	getProperties
wm.In.db	The following flow services are included:	
	createCopy	getManagers
	createDocument	getProfileDocCollection

Folder	Built-in Services	
	createFromTemplate	getProfileDocument
	FTDomainSearch	getReplicationInfo
	FTSearch	getURLHeaderInfo
	getACL	getView
	getAgent	getViews
	getAgents	open
	getAllDocuments	queryAccess
	getDocumentByID	recycle
	getDocumentByUNID	remove
	getDocumentByURL	search
	getForm	updateFTIndex
	getForms	
	The following Java services are included:	
	getDatabaseProperties	setDatabaseProperty
	getDatabaseProperty	
wm.ln.dbDirectory	The following flow services are included:	
	createDatabase	openDatabaseByReplicaID
	getFirstDatabase	openDatabaseIfModified
	getNextDatabase	openMailDatabase
	getServerName	recycle
	openDatabase	
wm.ln.document	The following flow services are included:	
	computeWithForm	getLastAccessedDateTime
	copyAllItems	getLastModifiedDateTime
	copyItem	getParentDatabase
	copyToDatabase	getParentView
	createReplyMessage	getResponses

Folder	Built-in Services	
	createRichTextItem	hasItem
	encrypt	makeResponse
	generateXML	putInFolder
	getAttachment	remove
	getAuthors	removeFromFolder
	getColumnValues	removeItem
	getCreatedDateTime	renderToRTItem
	getEmbeddedObjects	replaceItemValue
	getEncryptionKeys	save
	getFolderReferences	send
	getItems	setEncryptionKeys
	getItemValue	sign
	The following Java services are included:	
	appendItemsFromRecord	getProperties
	appendItemValue	setProperties
	getFirstItem	
wm.In.documentCollection	The following flow services are included:	
	addDocument	getPrevDocument
	deleteDocument	getQuery
	FTSearch	isSorted
	getCount	putAllInFolder
	getFirstDocument	recycle
	getLastDocument	removeAll
	getNextDocument	removeAllFromFolder
	getNthDocument	stampAll
	The following Java service is included:	
	hasDocument	

Folder	Built-in Services
wm.ln.embeddedObject	The following flow services are included:
	extractFile getReader
	getClassname getSource
	getFileSize getType
	getInputStream remove
	getName
wm.ln.form	The following flow services are included:
	getAliases recycle
	getFields remove
	getFormUsers setFormUsers
	getParent setReaders
	getReaders
	The following Java services are included:
	getFormProperties setFormProperties
	getProperties
wm.ln.item	The following flow services are included:
	addNewLine getFormattedText
	addPageBreak getInputStream
	addTab getLastModified
	appendDbLink getMIMEEntity
	appendDocLink getName
	appendParagraphStyle getParentDocument
	appendRTItem getReader
	appendStyle getText
	appendText getType
	appendViewLink getValueLength
	embedObject getValues

Folder	Built-in Services																																										
	<table border="1"> <tr> <td data-bbox="578 254 992 306">getDatetimeValue</td> <td data-bbox="992 254 1377 306">remove</td> </tr> <tr> <td data-bbox="578 306 992 359">getEmbeddedObject</td> <td data-bbox="992 306 1377 359">setDateTimeValue</td> </tr> <tr> <td colspan="2" data-bbox="578 359 1377 426">getEmbeddedObjects</td> </tr> </table>	getDatetimeValue	remove	getEmbeddedObject	setDateTimeValue	getEmbeddedObjects																																					
getDatetimeValue	remove																																										
getEmbeddedObject	setDateTimeValue																																										
getEmbeddedObjects																																											
wm.In.international	<p data-bbox="578 436 1081 468">The following Java service is included:</p> <table border="1"> <tr> <td data-bbox="578 489 1377 541">getProperties</td> </tr> </table>	getProperties																																									
getProperties																																											
wm.In.item	<p data-bbox="578 552 1114 583">The following flow services are included:</p> <table border="1"> <tr> <td data-bbox="578 604 992 657">addNewLine</td> <td data-bbox="992 604 1377 657">getFormattedText</td> </tr> <tr> <td data-bbox="578 657 992 709">addPageBreak</td> <td data-bbox="992 657 1377 709">getInputStream</td> </tr> <tr> <td data-bbox="578 709 992 762">addTab</td> <td data-bbox="992 709 1377 762">getLastModified</td> </tr> <tr> <td data-bbox="578 762 992 814">appendDbLink</td> <td data-bbox="992 762 1377 814">getMIMEEntity</td> </tr> <tr> <td data-bbox="578 814 992 867">appendDocLink</td> <td data-bbox="992 814 1377 867">getName</td> </tr> <tr> <td data-bbox="578 867 992 919">appendParagraphStyle</td> <td data-bbox="992 867 1377 919">getParentDocument</td> </tr> <tr> <td data-bbox="578 919 992 972">appendRTItem</td> <td data-bbox="992 919 1377 972">getReader</td> </tr> <tr> <td data-bbox="578 972 992 1024">appendStyle</td> <td data-bbox="992 972 1377 1024">getText</td> </tr> <tr> <td colspan="2" data-bbox="578 1024 1377 1077"> </td> </tr> <tr> <td data-bbox="578 1077 992 1129">appendText</td> <td data-bbox="992 1077 1377 1129">getType</td> </tr> <tr> <td data-bbox="578 1129 992 1182">appendViewLink</td> <td data-bbox="992 1129 1377 1182">getValueLength</td> </tr> <tr> <td data-bbox="578 1182 992 1234">embedObject</td> <td data-bbox="992 1182 1377 1234">getValues</td> </tr> <tr> <td data-bbox="578 1234 992 1287">getDatetimeValue</td> <td data-bbox="992 1234 1377 1287">remove</td> </tr> <tr> <td data-bbox="578 1287 992 1339">getEmbeddedObject</td> <td data-bbox="992 1287 1377 1339">setDateTimeValue</td> </tr> <tr> <td colspan="2" data-bbox="578 1339 1377 1392">getEmbeddedObjects</td> </tr> <tr> <td colspan="2" data-bbox="578 1392 1377 1444"> </td> </tr> <tr> <td colspan="2" data-bbox="578 1444 1377 1497">The following Java services are included:</td> </tr> <tr> <td data-bbox="578 1497 992 1549">getProperties</td> <td data-bbox="992 1497 1377 1549">setValue</td> </tr> <tr> <td data-bbox="578 1549 992 1602">getValue</td> <td data-bbox="992 1549 1377 1602">setValues</td> </tr> <tr> <td colspan="2" data-bbox="578 1602 1377 1654"> </td> </tr> <tr> <td colspan="2" data-bbox="578 1654 1377 1696">setProperties</td> </tr> </table>	addNewLine	getFormattedText	addPageBreak	getInputStream	addTab	getLastModified	appendDbLink	getMIMEEntity	appendDocLink	getName	appendParagraphStyle	getParentDocument	appendRTItem	getReader	appendStyle	getText			appendText	getType	appendViewLink	getValueLength	embedObject	getValues	getDatetimeValue	remove	getEmbeddedObject	setDateTimeValue	getEmbeddedObjects				The following Java services are included:		getProperties	setValue	getValue	setValues			setProperties	
addNewLine	getFormattedText																																										
addPageBreak	getInputStream																																										
addTab	getLastModified																																										
appendDbLink	getMIMEEntity																																										
appendDocLink	getName																																										
appendParagraphStyle	getParentDocument																																										
appendRTItem	getReader																																										
appendStyle	getText																																										
appendText	getType																																										
appendViewLink	getValueLength																																										
embedObject	getValues																																										
getDatetimeValue	remove																																										
getEmbeddedObject	setDateTimeValue																																										
getEmbeddedObjects																																											
The following Java services are included:																																											
getProperties	setValue																																										
getValue	setValues																																										
setProperties																																											
wm.In.mimeEntity	<p data-bbox="578 1707 1114 1738">The following flow services are included:</p> <table border="1"> <tr> <td data-bbox="578 1759 992 1812">getContentAsText</td> <td data-bbox="992 1759 1377 1812">getInputSource</td> </tr> <tr> <td data-bbox="578 1812 992 1864">getContentSubType</td> <td data-bbox="992 1812 1377 1864">getInputStream</td> </tr> </table>	getContentAsText	getInputSource	getContentSubType	getInputStream																																						
getContentAsText	getInputSource																																										
getContentSubType	getInputStream																																										

Folder	Built-in Services	
	getContenttype	getNextSibling
	getFirstChildEntity	getParentEntity
	getHeaders	getReader
wm.ln.name	The following Java service is included:	
	getProperties	
wm.ln.newsletter	The following flow services are included:	
	createNewsletter	isDoSubject
	formatDocument	setDoScore
	formatMsgWithDoclinks	setDoSubject
	getSubjectItemName	setSubjectItemName
	isDoScore	
wm.ln.replication	The following Java services are provided:	
	getProperties	setProperties
wm.ln.richText.richTextItem	The following flow services are included:	
	addNewLine	appendStyle
	addPageBreak	appendText
	addTab	appendViewLink
	appendDbLink	embedObject
	appendDocLink	getEmbeddedObject
	appendParagraphStyle	getEmbeddedObjects
	appendRTItem	getFormattedText
wm.ln.richText.richTextStyle	The following Java services are included:	
	getProperties	setProperties
wm.ln.richText.richTextTab	The following flow service is included:	
	clear	
	The following Java service is included:	
	getProperties	

Folder	Built-in Services																	
wm.ln.richText.rtParagraphStyle	<p>The following flow services are included:</p> <table border="1"> <tr> <td>clearAllTabs</td> <td>setTab</td> </tr> <tr> <td>getTabs</td> <td>setTabs</td> </tr> </table> <p>The following Java services are included:</p> <table border="1"> <tr> <td>getProperties</td> <td>setProperties</td> </tr> </table>	clearAllTabs	setTab	getTabs	setTabs	getProperties	setProperties											
clearAllTabs	setTab																	
getTabs	setTabs																	
getProperties	setProperties																	
wm.ln.session	<p>The following flow services are included:</p> <table border="1"> <tr> <td>createNewsletter</td> <td>getEnvironmentString</td> </tr> <tr> <td>createRichTextParagraphStyle</td> <td>getInternational</td> </tr> <tr> <td>createRichTextStyle</td> <td>getInternationalProperties</td> </tr> <tr> <td>evaluate</td> <td>getURLDatabase</td> </tr> <tr> <td>freeTimeSearch</td> <td>getUserNameObject</td> </tr> <tr> <td>getAddressBooks</td> <td>openDatabase</td> </tr> <tr> <td>getDatabase</td> <td>resolve</td> </tr> <tr> <td>getDbDirectory</td> <td>setEnvironmentString</td> </tr> </table> <p>The following Java service is included:</p> <table border="1"> <tr> <td>getProperties</td> </tr> </table>	createNewsletter	getEnvironmentString	createRichTextParagraphStyle	getInternational	createRichTextStyle	getInternationalProperties	evaluate	getURLDatabase	freeTimeSearch	getUserNameObject	getAddressBooks	openDatabase	getDatabase	resolve	getDbDirectory	setEnvironmentString	getProperties
createNewsletter	getEnvironmentString																	
createRichTextParagraphStyle	getInternational																	
createRichTextStyle	getInternationalProperties																	
evaluate	getURLDatabase																	
freeTimeSearch	getUserNameObject																	
getAddressBooks	openDatabase																	
getDatabase	resolve																	
getDbDirectory	setEnvironmentString																	
getProperties																		
wm.ln.utils.garbage	<p>The following Java service is included:</p> <table border="1"> <tr> <td>recycle</td> </tr> </table>	recycle																
recycle																		
wm.ln.utils.records	<p>This folder includes the following document:</p> <table border="1"> <tr> <td>mailrecord</td> </tr> </table>	mailrecord																
mailrecord																		
wm.ln.view	<p>The following flow services are included:</p> <table border="1"> <tr> <td>clear</td> <td>getLastDocument</td> </tr> <tr> <td>createViewNav</td> <td>getNextDocument</td> </tr> <tr> <td>createViewNavFromCategory</td> <td>getNextSibiling</td> </tr> <tr> <td>createViewNavFromChildren</td> <td>getNthDocument</td> </tr> <tr> <td>createViewNavFromDescendants</td> <td>getParentDocument</td> </tr> <tr> <td>createViewNavFromDocument</td> <td>getPrevDocument</td> </tr> </table>	clear	getLastDocument	createViewNav	getNextDocument	createViewNavFromCategory	getNextSibiling	createViewNavFromChildren	getNthDocument	createViewNavFromDescendants	getParentDocument	createViewNavFromDocument	getPrevDocument					
clear	getLastDocument																	
createViewNav	getNextDocument																	
createViewNavFromCategory	getNextSibiling																	
createViewNavFromChildren	getNthDocument																	
createViewNavFromDescendants	getParentDocument																	
createViewNavFromDocument	getPrevDocument																	

Folder	Built-in Services	
	createViewNavFromViewEntry	getPrevSibling
	createViewNavMaxLevel	getReaders
	FTSearch	refresh
	getAllEntries	remove
	getChildDocument	setAutoUpdate
	getColumn	setProtectReaders
	getFirstDocument	setReaders
	The following Java services are included:	
	getAliases	getDocumentByKey
	getAllDocumentsByKey	getEntryByKey
	getAllEntriesByKey	getProperties
	getColumnNames	getProperty
	getColumns	
wm.In.viewColumn	The following flow service is included:	
	getParentView	
	The following Java services are included:	
	getProperties	getProperty
wm.In.viewEntry	The following flow services are included:	
	getColumnValues	getParent
	getDocument	
	The following Java services are included:	
	getProperties	getProperty
wm.In.viewEntryCollection	The following flow services are included:	
	addEntry	getNthEntry
	deleteEntry	getParent
	FTSearch	getPrevEntry
	getCount	getQuery

Folder	Built-in Services	
	getEntry	putAllInFolder
	getFirstEntry	removeAll
	getLastEntry	removeAllFromFolder
	getNextEntry	stampAll
wm.In.viewNavigator	The following flow services are included:	
	getCacheSize	getPrevSibiling
	getChild	gotoChild
	getCurrent	gotoEntry
	getFirst	gotoFirst
	getFirstDocument	gotoFirstDocument
	getLast	gotoLast
	getLastDocument	gotoLastDocument
	getMaxLevel	gotoNext
	getNext	gotoNextCategory
	getNextCategory	gotoNextDocument
	getNextDocument	gotoNextSibiling
	getNextSibiling	gotoParent
	getNthEntry	gotoPos
	getParent	gotoPrev
	getParentView	gotoPrevCategory
	getPos	gotoPrevDocument
	getPrev	gotoPrevSibiling
	getPrevCategory	setCacheSize
	getPrevDocument	setMaxLevel

Built-In Services Reference

This section lists all built-in flow and Java services provided with Lotus Notes Adapter. The services are listed alphabetically. However, parameter information is included only for those services that

provide additional processing value. For information about the inputs and outputs used by the wrapper services, see the documentation for the Lotus Domino Toolkit for Java/CORBA for the corresponding `lotus.domino` class and method.

wm.In.viewNavigator:getNextSibling

This flow service returns the next sibling entry after the current or specified entry in a view navigator.

Input Parameters

<i>viewNavigator</i>	ViewNavigator The ViewNavigator object to get the next sibling from.
<i>entry</i>	ViewEntry Optional. An entry in the view. Cannot be null.

Output Parameters

<i>nextSiblingViewEntry</i>	ViewEntry The sibling entry following the current or specified entry. Returns null if there is no next sibling entry.
-----------------------------	--

wm.In.acl:aclEntryRecycle

This flow service unconditionally destroys an ACLentry object and returns its memory to the system.

Input Parameters

<i>aclEntry</i>	ACLEntry The <code>lotus.domino.ACLEntry</code> object to recycle.
-----------------	---

wm.In.acl:aclRecycle

This flow service unconditionally destroys an ACL object and returns its memory to the system.

Input Parameters

<i>acl</i>	ACL The <code>lotus.domino.ACL</code> object to recycle.
------------	---

wm.In.acl:addRole

This flow service adds a role to an ACL.

Input Parameters

<i>roleName</i>	String The role to add to the lotus.domino.ACL object. Square brackets are not required around the roleName.
<i>Acl</i>	ACL The lotus.domino.ACL object to add the role to.

Output Parameters

<i>value</i>	String "SUCCESS" if the addRole method in the lotus.domino.ACL class executes without problems.
--------------	--

wm.In.acl:createACLEntry

This flow service creates an entry in the ACL with the name and level specified. The ACL.save() should be called explicitly to save the modified ACL to disk.

Input Parameters

<i>acl</i>	ACL The lotus.domino.ACL object to create the ACL entry in.
<i>name</i>	String The name of the user, group, or server for whom you want to create an entry in the ACL. You must supply the complete name but hierarchical names can be in abbreviated format. Case is not significant.
<i>level</i>	String The level you want to assign to this person, group, or server in the ACL. <ul style="list-style-type: none">■ LEVEL_NOACCESS■ LEVEL_DEPOSITOR■ LEVEL_READER■ LEVEL_AUTHOR■ LEVEL_EDITOR■ LEVEL_DESIGNER■ LEVEL_MANAGER

Output Parameters

<i>aclEntry</i>	ACLEntry The lotus.domino.ACLEntry object created.
-----------------	---

wm.In.acl:deleteRole

This flow service deletes a role from an ACL and removes it from any ACL entries that have it. The ACL.save() must be explicitly called to save the modified ACL to disk.

Input Parameters

<i>roleName</i>	String The name of the role to delete. Square brackets are not required around the name.
-----------------	---

<i>acl</i>	ACL The lotus.domino.ACL object to delete the role from.
------------	---

Output Parameters

<i>value</i>	String "SUCCESS" if the deleteRole method in the lotus.domino.ACL class executes without problems.
--------------	---

wm.In.acl:disableRole

This flow service disables a specified role in an ACLEntry. The role must exist in the ACLEntry. If the role exists, but is already disabled for the entry, this method does nothing. The ACL.save() must be called explicitly to save the modified ACL to disk.

Input Parameters

<i>aclEntry</i>	ACLEntry The lotus.domino.ACLEntry object to disable the role in.
-----------------	--

<i>roleName</i>	String The name of the role to disable in the given ACLEntry. Square brackets are not required around the <i>roleName</i> .
-----------------	--

Output Parameters

<i>value</i>	String "SUCCESS" if the disableRole method in the lotus.domino.ACLEntry class executes without problems.
--------------	---

wm.In.acl:enableRole

This flow service enables a given role for an ACLEntry. The role must exist in the ACL. If the role exists, but is already enabled for the entry, this service does nothing. The ACL.save() must be explicitly called to save the modified ACL to disk.

Input Parameters

<i>roleName</i>	String The name of the role to enable in the given ACLEntry. Square brackets are not required around the <i>roleName</i> .
<i>aclEntry</i>	ACLEntry The lotus.domino.ACLEntry object to enable the role in.

Output Parameters

<i>value</i>	String "SUCCESS" if the enableRole method in the lotus.domino.ACLEntry class executes without problems.
--------------	--

wm.In.acl:getACLEntryProperties

This Java service returns the properties of an ACLEntry object.

Input Parameters

<i>aclEntry</i>	ACLEntry The lotus.domino.ACLEntry object to obtain the properties of.
-----------------	---

Output Parameters

<i>properties</i>	Record A record containing the lotus.domino.ACLEntry properties: <ul style="list-style-type: none">■ level■ name■ nameObject■ parent■ userType■ isAdminReaderAuthor
-------------------	---

- isAdminServer
 - canCreateDocuments
 - canDeleteDocuments
 - canCreatePersonalFolder
 - canCreatePersonalAgent
 - canCreateSharedFolder
 - canCreateLSorJavaAgent
 - isGroup
 - isPerson
 - isPublicReader
 - isPublicWriter
 - isServer
-

wm.In.acl:getACLEntryProperty

This Java service returns the value of a specified property in an `ACLEntry` object.

Input Parameters

<i>property</i>	String The name of the property to obtain.
<i>aclEntry</i>	ACLEntry The <code>lotus.domino.ACLEntry</code> object to obtain the specified property from.

Output Parameters

<i>value</i>	String The value of the specified property.
--------------	--

wm.In.acl:getACLEntryRoles

This flow service fetches the roles associated with a specified `ACLEntry` object.

Input Parameters

<i>aclEntry</i>	ACLEntry The <code>lotus.domino.ACLEntry</code> object to fetch roles in.
-----------------	--

Output Parameters

roles **String List** A list of the roles.

wm.In.acl:getACLProperties

This Java service returns the properties of a specified ACL object.

Input Parameters

acl **ACL** The lotus.domino.ACL object to obtain the properties of.

Output Parameters

properties **Record** A record containing the lotus.domino.ACL properties.

wm.In.acl:getACLProperty

This Java service returns the value of a certain property in a specified ACL object.

Input Parameters

property **String** The name of the property to obtain.

acl **ACL** The lotus.domino.ACL object to obtain the specified property from.

Output Parameters

value **String** The value of the requested property.

wm.In.acl:getACLRoles

This flow service fetches the roles associated with a specified ACL object.

Input Parameters

acl **ACL** The lotus.domino.ACL object to fetch roles in.

Output Parameters

<i>roles</i>	String List A list of the roles.
--------------	---

wm.In.acl:getEntry

This flow service finds an ACLEntry in an ACL. People, groups, or servers in an ACL can be found. If a person is not listed explicitly in the ACL, but is a member of a group listed in the ACL, `getEntry` does not find that person's name; use the `Database.queryAccess` method instead.

Input Parameters

<i>acl</i>	ACL The lotus.domino.ACL object to find an entry in.
<i>aclName</i>	String The name whose ACL entry is to be found. The complete names must be supplied but hierarchical names can be in abbreviated format. Case is not significant.

Output Parameters

<i>aclEntry</i>	ACLEntry The ACL entry that matches the name. If <i>aclName</i> is not in the ACL, returns null.
-----------------	---

wm.In.acl:getFirstEntry

This flow service returns the first entry in an ACL. The first entry is typically the Default entry.

Input Parameters

<i>acl</i>	ACL The lotus.domino.ACL object to get the first entry from.
------------	---

Output Parameters

<i>aclEntry</i>	ACLEntry The first lotus.domino.ACLEntry object in the provided ACL object.
-----------------	--

wm.In.acl:getNextEntry

This flow service returns the next ACL entry or the one following a specified entry. With no parameter, this method returns the entry following the last entry retrieved. This no-parameter method improves performance for remote operations because ACL entries are cached locally.

Input Parameters

acl **ACL** The lotus.domino.ACL object to get the next entry from.

aclEntry **ACLEntry** Optional. The lotus.domino.ACLEntry object previous to the object to be fetched.

Output Parameters

aclEntry **ACLEntry** The entry following the last entry retrieved or specified.

wm.In.acl:isRoleEnabled

This flow service checks if the role specified is enabled. The role must exist in the ACL.

Input Parameters

roleName **String** The name of the role to probe.

aclEntry **ACLEntry** The lotus.domino.ACLEntry object where the role is to be checked.

Output Parameters

value **String** True if the role is enabled for this entry, false if the role is not enabled for this entry.

wm.In.acl:remove

This flow service removes an entry from an access control list. The ACL.save() method must be explicitly called to save the changes.

Input Parameters

aclEntry **ACLEntry** The lotus.domino.ACLEntry object to remove the role from.

Output Parameters

<i>value</i>	String "SUCCESS" if the remove method in the lotus.domino.ACLEntry class executes without problems.
--------------	--

wm.In.acl:removeACLEntry

This flow service removes an entry from an ACL. Execute the ACL.save() method on the ACL object in order for the changes to be saved to the database.

Input Parameters

<i>acl</i>	ACL The lotus.domino.ACL object to remove the role in.
<i>roleName</i>	String The name of the person, group, or server whose entry you want to remove. Supply the complete name but hierarchical names can be in abbreviated format. Case is not significant.

Output Parameters

<i>value</i>	String "SUCCESS" if the removeACLEntry method in the lotus.domino.ACL class executes without problems.
--------------	---

wm.In.acl:renameRole

This flow service changes the name of a role.

Input Parameters

<i>acl</i>	ACL The lotus.domino.ACLEntry object to rename a role in.
<i>oldName</i>	String The current name of the role. Square brackets not required around the name.
<i>newName</i>	String The new name to be given to the role. Square brackets are not required around the name.

wm.In.acl:save

This flow service saves changes made to an ACL. If it is not called before closing a database, the changes made to its ACL are lost.

Input Parameters

<i>acl</i>	ACL The lotus.domino.ACL object to save changes to.
------------	--

wm.In.acl:setACLEntryProperty

This Java service sets the properties of an ACLEntry object.

Input Parameters

<i>aclEntry</i>	ACLEntry The lotus.domino.ACLEntry object to set a property in.
-----------------	--

<i>property</i>	String The name of the property to set.
-----------------	--

<i>value</i>	String The value of the property to set.
--------------	---

wm.In.acl:setACLProperty

This Java service sets the properties of an ACL object.

Input Parameters

<i>acl</i>	ACL The lotus.domino.ACL object to set a property in.
------------	--

<i>property</i>	String The value of the property to set.
-----------------	---

<i>value</i>	String The value of the property to set.
--------------	---

wm.In.agent:getLastRunDateTime

This flow service gets the date that an agent last ran. If the script has never been run before, this property returns null.

Input Parameters

<i>agent</i>	Agent The lotus.domino.Agent object to inspect.
--------------	--

Output Parameters

lastRunDateTime **DateTime** A lotus.domino.DateTime object for the last run date of the Agent.

wm.In.agent:getParent

This flow service gets the parent database that contains the agent.

Input Parameters

agent **Agent** The lotus.domino.Agent object to get the database parent of.

Output Parameters

database **Database** The lotus.domino.Database parent.

wm.In.agent:getProperties

This Java service gets the properties of an agent.

Input Parameters

agent **Agent** The lotus.domino.Agent object to get properties for.

Output Parameters

properties **Record** A record containing the properties for the lotus.domino.Agent object.

- name
- owner
- isEnabled
- serverName
- query
- comment
- isPublic

- commonOwner
 - trigger
 - target
 - isNotesAgent
 - isWebAgent
 - URL
 - parameterDocID
-

wm.In.agent:getProperty

This Java service returns the value of a certain property in an Agent object.

Input Parameters

<i>agent</i>	Agent The lotus.domino.Agent object to obtain the specified property from.
<i>property</i>	String The name of the property to obtain.

Output Parameters

<i>value</i>	String The value of the specified property.
--------------	--

wm.In.agent:remove

This flow service permanently deletes an agent from a database. Once you call this method, the Agent object is null and you can no longer use its methods or properties. It is possible to use this method to delete the agent that is currently running. Use this feature with care. You must refresh the Agent's view in the user interface to see that the agent has been removed.

Input Parameters

<i>agent</i>	Agent The lotus.domino.Agent object to remove.
--------------	---

wm.In.agent:run

This flow service runs the agent. You cannot run an agent recursively (cannot call it from itself). The user cannot interact directly with a called agent. User output goes to the Domino server log. You cannot debug a called agent.

Input Parameters

<i>agent</i>	Agent The lotus.domino.Agent object to run.
<i>NoteID</i>	String Optional. The NoteID of a document. The value is passed to the ParameterDocID property of the called agent.

wm.In.agent:runOnServer

This flow service runs the agent on the computer containing the database.

Input Parameters

<i>agent</i>	Agent The lotus.domino.Agent object to run.
<i>NoteID</i>	String Optional. The NoteID of a document. The value is passed to the ParameterDocID property of the called agent.

Output Parameters

<i>Status</i>	String Status of the operation, where 0 indicates success.
---------------	---

wm.In.authentication:login

This flow service is a place holder for users of the adapter upgrading from previous versions. For information about changes to this service when upgrading from Lotus Notes Adapter 1.0.x to Lotus Notes Adapter 6.0, see [“Upgrading to Lotus Notes Adapter 6.0” on page 23](#).

wm.In.authentication:logout

This Java service invalidate a session in Lotus Notes. The service will also recycle any Domino objects at the root level of the pipeline. For more information, see [“Freeing Resources After Processing” on page 16](#).

Input Parameters

<i>session</i>	Session The lotus.domino.Session to recycle.
----------------	---

Output Parameters

<i>success</i>	String "false" if the session object was missing, "true" if the session object was found and the recycle process completed without errors.
<i>recycledObject</i>	String List A list of recycled objects found at the root level of the pipeline.

wm.In.dateRange:createDR

This Java service creates a date range object for the given start date and end date specified.

Input Parameters

<i>session</i>	Session The lotus.domino.Session object to create the date with.
<i>startDate</i>	DateObject, DateTime, and String Optional. Can be java.util.Date, lotus.domino.DateTime or java.lang.String in "MM/dd/yy hh:mm:ss AM/PM" format.
<i>endDate</i>	DateObject, DateTime, and String Optional. Can be java.util.Date, lotus.domino.DateTime or java.lang.String in "MM/dd/yy hh:mm:ss AM/PM" format.

Output Parameters

<i>dateRange</i>	DateRange A lotus.domino.DateRange object created based on the provided input.
------------------	---

wm.In.dateRange:createDRFromString

This flow service returns a date range object for the given date string.

Input Parameters

<i>session</i>	Session The lotus.domino.Session object to create the date with.
<i>drString</i>	String The format for the date range. The format for <i>drString</i> should be "mm/dd/yy hh:mm:ss AM/PM -mm/dd/yy hh:mm:ss AM/PM"

Output Parameters

<i>drString</i>	String The Text property of the DateRange object.
-----------------	--

wm.In.dateRange:setEndDateTime

This flow service sets the end date of a date range.

Input Parameters

<i>dateRange</i>	DateRange The lotus.domino.DateRange object to set the end date for.
------------------	---

<i>endDateTime</i>	DateTime The lotus.domino.DateTime object representing the end of the date range.
--------------------	--

wm.In.dateRange:setStartDateTime

This flow service sets the start date of a date range.

Input Parameters

<i>dateRange</i>	DateRange The lotus.domino.DateRange object to set the start date for.
------------------	---

<i>startDateTime</i>	DateTime The lotus.domino.DateTime object representing the start of the date range.
----------------------	--

wm.In.dateTime:createDT

This flow service creates a lotus.domino.DateTime object given some parameters as string inputs.

Input Parameters

<i>session</i>	Session The lotus.domino.Session object to create the lotus.domino.DateTime object with.
----------------	---

<i>month</i>	String 2-digit month.
--------------	------------------------------

<i>day</i>	String 2-digit day.
------------	----------------------------

<i>year</i>	String 4-digit year.
-------------	-----------------------------

<i>hour</i>	String 2-digit hour.
-------------	-----------------------------

<i>minute</i>	String 2-digit minute.
<i>second</i>	String 2-digit second.
<i>AMPM</i>	String "AM" or "PM".

Output Parameters

<i>dateTime</i>	DateTime The lotus.domino.DateTime object created.
-----------------	---

wm.In.dateTime:createDT_old

This Java service does not perform functionality specific to Lotus Notes. It is used to create a java.util.Date object in a similar fashion to wm.In.dateTime:createDT.

wm.In.dateTime:createDTFromDate

This flow service creates a lotus.domino.DateTime object from a java.util.Date object.

Input Parameters

<i>session</i>	Session The lotus.domino.Session object to create the lotus.domino.DateTime object with.
<i>date</i>	String The java.util.Date object to create the lotus.domino.DateTime object with.

Output Parameters

<i>dateTime</i>	DateTime The lotus.domino.DateTime object created.
-----------------	---

wm.In.dateTime:createDTFromString

This flow service creates a lotus.domino.DateTime object from a String.

Input Parameters

<i>session</i>	Session The lotus.domino.Session object to create the lotus.domino.DateTime object with.
<i>dateString</i>	String Date string in the "mm/dd/yy hh:mm:ss AM/PM" format.

Output Parameters

dateTime **DateTime** The lotus.domino.DateTime object created.

wm.In.dateTime:getProperties

This Java service gets properties of the DateTime object.

Input Parameters

dateTime **DateTime** The lotus.domino.DateTime object to get properties for.

Output Parameters

Properties **Record** A record containing:

- DateOnly
- GMTTime
- IsDST
- LocalTime
- TimeOnly
- TimeZone
- ZoneTime

wm.In.db:createCopy

This flows service creates a copy of a database. Programs running on a server or making remote (IIOP) calls to a server cannot create or access databases on other servers.

Input Parameters

database **Database** The lotus.domino.Database object to make a copy from.

dbFileName **String** The file name of the new copy. If a database with the specified file name already exists, an exception is thrown.

<i>DateRange</i>	DateRange Optional. The maximum size (in gigabytes) to assign to the new database. This parameter applies only to Release 4 databases or those created on a server that has not been upgraded to Release 5. Entering an integer greater than 4, generates a run-time error.
------------------	--

Output Parameters

<i>copiedDatabase</i>	Database An object representing the new created database.
-----------------------	--

wm.In.db:createDocument

This flow service creates a document in a database and returns a Document object that represents the new document. You must call Document.save() if you want the new document to be saved.

Input Parameters

<i>database</i>	Database The lotus.domino.Database object for the database to create the document in.
-----------------	--

Output Parameters

<i>document</i>	Document The created lotus.domino.Document object.
-----------------	---

wm.In.db:createFromTemplate

This flow service creates a new database from an existing database. If the "Database is a template" property is selected, then the new database has the design elements and documents of the existing database. If the property is not selected, the source is treated as a database instead of a template, and does not convert entries such as [-Default-] in the ACL to regular ACL entries in the newly created database. Programs running on a server or making remote (IIOP) calls to a server cannot create or access databases on other servers.

Input Parameters

<i>dbFileName</i>	String The file name of the new database. If a database with the specified file name already exists, an exception is thrown.
-------------------	---

inherit **String** Specify "true" if you want the new database to inherit future design changes from the template; otherwise, specify "false".

templateDb **Database** The template database object.

Output Parameters

newdatabase **Database** An object representing the created database.

wm.In.db:FTDomainSearch

This flow service conducts a Domain Search, that is, a full-text search of all databases listed in a Domain Catalog and marked as included for multi-database indexing. If you do not specify any sort options, you get the results sorted by relevance score. If you specify a sort by date, you do not get relevance scores.

Input Parameters

query **String** The full-text query.

max **String** The maximum number of results you want returned from the query. Set this parameter to 0 to receive all matching results.

sortopt **String** Use one of the following constants to specify a sorting option:

- FT_SCORES -(default) sorts by relevance score. When the collection is sorted by relevance, the highest relevance appears first. To access the relevance score of each document in the collection, use the FTSearchScore property in returned Documents.
 - FT_DATE_DES - sorts by document creation date in descending order.
 - FT_DATE_ASC - sorts by document creation date in ascending order.
-

otheropt **String** Use the following constants to specify additional search options. To specify more than one option, use a logical or operation: #

- FT_DATABASE - includes Domino databases in the search scope.

- FT_FILESYSTEM - includes files other than Domino databases in the search scope.
- FT_FUZZY - specifies a fuzzy search.
- FT_STEMS - uses stem words as the basis of the search.

<i>start</i>	String The starting page to return.
<i>count</i>	String The number of pages to return.
<i>entryForm</i>	String The name of the search form in the domain catalog, for example, "Domain Search."
<i>database</i>	Database The current lotus.domino.Database object must represent a Domain Catalog.

Output Parameters

<i>document</i>	Document A document that contains the results of the full-text query.
-----------------	--

wm.In.db:FTSearch

This flow service conducts a full-text search of all the documents in a database. If the database is not full-text indexed, this method works, but less efficiently.

To test for an index, use the `IsFTIndexed` property. To create an index on a local database, use the `updateFTIndex` method.

FTSearch searches all documents in a database. To search only documents found in a particular view, use the FTSearch method in View. To search only documents found in a particular document collection, use the FTSearch method in DocumentCollection.

If you do not specify any sort options, you get the documents sorted by relevance score. If you sort by date, you do not get relevance scores.

A Newsletter object formats its doclink report with either the document creation date or the relevance score, depending on the sort options you use in the document collection. If the database has a multi-database index, you get a multi-database search. Navigation through the resulting document collection may be slow, but you can create a newsletter from the collection.

Input Parameters

<i>query</i>	String The full-text query.
--------------	------------------------------------

<i>max</i>	String Optional. The maximum number of results you want returned from the query. Set this parameter to 0 to receive all matching results.
<i>sortopt</i>	String Optional. If used, <i>otheropt</i> must also be used. Use one of the following constants to specify a sorting option: <ul style="list-style-type: none">■ FT_SCORES - (default) sorts by relevance score. When the collection is sorted by relevance, the highest relevance appears first. To access the relevance score of each document in the collection, use the <code>FTSearchScore</code> property in returned Documents.■ FT_DATE_DES - sorts by document creation date in descending order.■ FT_DATE_ASC - sorts by document creation date in ascending order.
<i>otheropt</i>	String Use the following constants to specify additional search options. To specify more than one option, use a logical or operation: # <ul style="list-style-type: none">■ FT_DATABASE - includes Domino databases in the search scope.■ FT_FILESYSTEM - includes files other than Domino databases in the search scope.■ FT_FUZZY - specifies a fuzzy search.■ FT_STEMS - uses stem words as the basis of the search.

Output Parameters

<i>documentCollection</i>	DocumentCollection A collection of documents that match the full-text query, sorted by the selected option. If no matches are found, the collection has a count of 0.
---------------------------	--

wm.ln.db:getACL

This flow service gets the access control list for a database.

Output Parameters

documentCollection

DocumentCollection A

lotus.domino.DocumentCollection object representing the documents in the database indicated.

wm.In.db:getDatabaseProperties

This Java service gets the database properties. Properties `createdDate`, `lastModifiedDate` and `lastFTIndexed` return variables of type `Date` objects.

Input Parameters

database

Database The `lotus.domino.Database` to get the properties for.

Output Parameters

properties

Record A record containing:

- title
- size
- server
- fileName
- filePath
- templateName
- designTemplateName
- isOpen
- categories
- accessLevel
- createdDate
- lastModifiedDate
- percentUsed
- replicaID
- getSizeQuota

- isDelayUpdates
- isFTIndexed
- isMultiDBSearch
- isPrivateAddressBook
- isPublicAddressBook
- lastFTIndexed
- maxSize
- folderReferencesEnabled
- parent

wm.In.db:getDatabaseProperty

This Java service gets a single property of a database.

Input Parameters

<i>property</i>	String The property name.
<i>database</i>	Database The lotus.domino.Database object to get a property from.

Output Parameters

<i>value</i>	Object The value of the property.
--------------	--

wm.In.db:getDocumentByID

This flow service fetches a document in a database, given the document note ID.

Input Parameters

<i>database</i>	Database The lotus.domino.Database object to get a document from.
<i>noteId</i>	String The note ID of the document to fetch.

Output Parameters

<i>document</i>	Document The document whose note ID matches the parameter.
-----------------	---

wm.In.db:getDocumentByUNID

This flow service finds a document in a database, given the document universal ID (UNID).

Input Parameters

<i>database</i>	Database The lotus.domino.Database object to get a document from.
-----------------	--

<i>unid</i>	String The universal ID of the document to fetch.
-------------	--

Output Parameters

<i>document</i>	Document The document whose universal ID matches the <i>unid</i> input.
-----------------	--

wm.In.db:getDocumentByUrl

This flow service finds a document in the Web Navigator database. You can use this method for either the Server Web Navigator or Personal Web Navigator databases. If you specify any of the optional parameters you must specify them all.

Input Parameters

<i>database</i>	Database The lotus.domino.Database object to get a document from.
-----------------	--

<i>url</i>	String The desired uniform resource locator (URL), for example, http://www.webmethods.com . Specify the entire URL starting with http. You can enter a maximum string length of 15K.
------------	---

<i>reload</i>	String True to reload the page from its Internet server. False to load the page from the Internet only if it is not already in the Web Navigator database.
---------------	---

<i>reloadIfModified</i>	String Optional. (Defaults to the specification for the second parameter, reload.) True to reload the page only if it has been modified on its Internet server.
-------------------------	--

False to load the page from the Internet only if it is not already in the Web Navigator database.

urlList

String Optional. (Defaults to false.) Web pages can contain URL links to other Web pages. You can specify whether to save the URLs in a field called URLLinks in the Notes document. (The Web Navigator creates a new URLLinksn field each time the field size reaches 64K. For example, the first URLLinks field is URLLinks1, the second is URLLinks2, and so on.) Specify true if you want to save the URLs in the URLLinksn fields. Specify false if you do not want to save the URLs in the URLLinksn fields. If you save the URLs, you can use them in agents. For example, you can create an agent that opens Web pages in the Web Navigator database and then loads all the Web pages saved in each of the URLLinksn fields.

Important:

Saving URLs in the URLLinksn fields may affect performance.

charSet

String Optional. (Defaults to null.) Enter the MIME character set (for example, ISO-2022-JP for Japanese or ISO-8859-1 for United States) that you want the Web Navigator to use when processing the Web page.

webUser

String Optional. (Defaults to null.) Some Internet servers require you to obtain a username before you can access their pages. This parameter allows you to enter the username that you previously obtained from the Full-text server.

webPasswd

String Optional. (Defaults to null.) Some Full-text servers require you to obtain a password before you can access their pages. This parameter allows you to enter the password that you previously obtained from the Internet server.

ProxyUser

String Optional. (Defaults to null.) Some proxy servers require that you specify a username in order to connect through them. This parameter allows you to enter the username for the proxy server. See your administrator for the username required by the proxy.

proxyPasswd

String Optional. (Defaults to null.) Some proxy servers require that you specify a password in order to connect through them. This parameter allows you to enter the password for the proxy server. See your administrator for the password required by the proxy.

noWait

String Optional. (Defaults to false.) Specify true to return immediately and not wait for completion of the retrieval. If you specify true, `getDocumentByUrl` does not return the Document object representing the URL document. This parameter is useful for offline storage purposes; in this case, you do not need the Document object and do not have to wait for completion of the operation.

Output Parameters

document

Document The Document that represents the URL document you specified.

wm.In.db:getForm

This flow service finds a form in a database, given the form name.

Input Parameters

database

Database The `lotus.domino.Database` object to get the form from.

name

String The name of the form to get.

Output Parameters

form

Form The `lotus.domino.Form` object representing the form fetched.

wm.In.db:getForms

This flow service gets all the forms in a database as Form object list.

Input Parameters

database

Database The `lotus.domino.Database` object to get the forms from.

Output Parameters

<i>form</i>	Form A list of lotus.domino.Form objects representing the forms in the database.
-------------	---

wm.In.db:getManagers

This flow service gets the people, servers, and groups that have Manager access to a database.

Input Parameters

<i>database</i>	Database The lotus.domino.Database object to get the managers from.
-----------------	--

Output Parameters

<i>manager</i>	String[] A list of managers.
----------------	--------------------------------------

wm.In.db:getProfileDocCollection

This flow service retrieves the profile documents associated with a profile form.

Input Parameters

<i>database</i>	Database The lotus.domino.Database object to get the documents from.
-----------------	---

<i>profileName</i>	String The name of the profile to fetch.
--------------------	---

Output Parameters

<i>documentCollection</i>	DateTime A lotus.domino.DocumentCollection object representing the documents in the profile indicated. If the profile has no documents and empty collection is returned.
---------------------------	---

wm.In.db:getProfileDocument

This flow service retrieves or creates a profile document.

Input Parameters

<i>database</i>	Database The lotus.domino.Database object to get the document and profile from.
<i>profile</i>	String The name or an alias of the profile form.
<i>profileUser</i>	String The user name (key) attached to the profile document.

Output Parameters

<i>document</i>	Document The document associated with the indicated profile.
-----------------	---

wm.In.db:getReplicationInfo

This flow service gives the replication object associated with this database. Each NotesDatabase instance contains one NotesReplication object. This object is used to get and set the replication setting of the server using the Replication class.

Input Parameters

<i>database</i>	Database The lotus.domino.Database object to get the replication object from.
-----------------	--

Output Parameters

<i>replication</i>	Replication The lotus.domino.Replication object associated with the provided database.
--------------------	---

wm.In.db:getURLHeaderInfo

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.db:getView

This flow service finds a view or folder in a database, given the name or alias of the view or folder. If the database is on a server, getView returns only public views and folders. When specifying the parameter, do not combine the view name and an alias. For example, specifying "By Author|AuthorView" does not work. Use either the view name ("By Author") or its alias ("AuthorView").

Input Parameters

<i>name</i>	String The case-insensitive name of a view or folder in a database. Use either the entire name of the view or folder (including backslashes for cascading views and folders), or an alias.
<i>database</i>	Database The lotus.domino.Database object to get the view object from.

Output Parameters

<i>view</i>	DateTime The view or folder whose name or synonym matches the parameter.
-------------	---

wm.In.db:getViews

This flow service sets the views and folders in a database as a View objects list. Each element of the object list represents a public view or folder in the database. If the database is local, personal folders are also included.

Input Parameters

<i>database</i>	Database The lotus.domino.Database to get the views from.
-----------------	--

Output Parameters

<i>view</i>	View[] A list of lotus.domino.View objects representing the view in the database.
-------------	---

wm.In.db:open

This flow service opens a database. Returns true if the database exists and is opened or false if no database with this name exists. A database must be open to use the Database properties and methods with some exceptions. An error is returned if the user does not have access rights to the database or server.

Input Parameters

<i>database</i>	Database The lotus.domino.Database representing the database to open.
-----------------	--

Output Parameters

isOpen **String** "true" is the operation succeeds, "false" otherwise.

wm.In.db:queryAccess

This flow service returns a person's, group's, or server's current access level to a database. Using this method at the same time an ACL object is in use may produce inconsistent results.

If the name you specify is not listed explicitly in the ACL, queryAccess does the following:

- If the name is a member of a group defined in the Domino Directory and that group name is listed in the ACL, queryAccess returns the access level of the group name.
- If the name is a member of several groups listed in the ACL, queryAccess returns the highest access level granted to any of the groups.
- If the name is not a member of any group, queryAccess returns the default access level.
- If the name you specify is listed explicitly in the ACL and is also a member of one or more groups listed in the ACL, queryAccess returns whichever access level is higher.

To see if a name is a member of a group, queryAccess uses the Domino Directories and Personal Address Books known to the computer on which the program is running. If the program is running on a workstation, queryAccess checks the Personal Address Book on the local computer. If the program is running on a server, queryAccess checks the Domino Directory on that server.

Input Parameters

user **String** The name of the person, group, or server. For a hierarchical name, the full name must be specified but can be in abbreviated format.

database **Database** The lotus.domino.Database object to query the access of a user in.

Output Parameters

accessLevel **String** The current access level, which is one of the following:

- ACL.LEVEL_AUTHOR
- ACL.LEVEL_DEPOSITOR
- ACL.LEVEL_DESIGNER
- ACL.LEVEL_EDITOR

- ACL.LEVEL_MANAGER
- ACL.LEVEL_NOACCESS
- ACL.LEVEL_READER

wm.In.db:recycle

This flow service unconditionally destroys an object and returns its memory to the system.

Input Parameters

<i>database</i>	Database The lotus.domino.Database object to recycle.
-----------------	--

wm.In.db:remove

This flow service permanently deletes a database.

Input Parameters

<i>database</i>	Database The lotus.domino.Database object to remove.
-----------------	---

wm.In.db:search

This flow service, given selection criteria for a document, returns all documents in a database that meet the criteria.

Input Parameters

<i>database</i>	Database The lotus.domino.Database object representing the database to search.
<i>formula</i>	String A Notes function formula that specifies the selection criteria.
<i>dateTime</i>	DateTime Optional. A cutoff date. The method searches only documents created or modified since the cutoff date. Can be null to indicate no cutoff date.
<i>max</i>	String Optional. Must specify the <i>dateTime</i> parameter to use this parameter. The maximum

number of documents you want returned.
Specify 0 to receive all matching documents.

Output Parameters

documentCollection

DocumentCollection An unsorted collection of documents that match the selection criteria.

wm.In.db:setDatabaseProperty

This Java service sets a property for the given database object.

Input Parameters

database

Database The lotus.domino.Database object to set the property in.

property

String The name of the property to set.

value

String The name of the property to set.

wm.In.db:updateFTIndex

This flow service updates the full-text index of a database.

Input Parameters

database

Database The lotus.domino.DateRange object.

create

String Specify true if you want to create an index if none exists (valid only for local databases). Otherwise, specify false.

wm.In.dbDirectory:createDatabase

This flow service creates a new database, using the file name that you specified. Because the new database is not based on a template, it is blank and does not contain any forms or views. If the database is open, only a subset of its methods are available.

Input Parameters

<i>dbDirectory</i>	DbDirectory The lotus.domino.DbDirectory object representing the directory to create the database in.
<i>dbFileName</i>	String The name of the new database.
<i>OpenDB</i>	String Optional. Indicates if you want to open the database. Specify true if you want to open the database and false if you do not.

Output Parameters

<i>database</i>	Database The created database.
-----------------	---------------------------------------

wm.In.dbDirectory:getFirstDatabase

This flow service returns the first database from a server or the local directory, using the file type specified. The returned database is closed. If the database is not opened only a subset of its methods are available. Each time this method is called, the database directory is reset and a new search is conducted. If template files are being searched, for example, a new call to `getFirstDatabase` with the *dbType* parameter set to DATABASE starts searching the directory from the beginning, this time for database files.

Input Parameters

<i>dbDirectory</i>	DbDirectory The lotus.domino.DbDirectory object representing the directory where the first database is to be fetched.
<i>dbType</i>	<p>String Indicates the kind of database file you want to retrieve:</p> <ul style="list-style-type: none"> ■ DbDirectory.DATABASE means any database (.nsf, .nsg, or .nsh file) ■ DbDirectory.TEMPLATE means any template (.ntf file) ■ DbDirectory.REPLICA_CANDIDATE means any database or template not disabled for replication ■ DbDirectory.TEMPLATE_CANDIDATE means any database or template

Output Parameters

database

Database The first database of the specified file type located in the directory, or null if the directory contains none.

wm.In.dbDirectory:getNextDatabase

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.dbDirectory:getServerName

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.dbDirectory:openDatabase

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.dbDirectory:openDatabaseByReplicaID

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.dbDirectory:openDatabaseSelfModified

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.dbDirectory:openMailDatabase

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.dbDirectory:recycle

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:appendItemsFromRecord

This Java service appends items to a document. If the items in the form are known, a record containing the items can be designed and given as input to be appended in the document.

Input Parameters

<i>document</i>	Document The lotus.domino.Document to append items to.
<i>record</i>	Record A record containing the items to append.

wm.In.document:appendItemValue

This Java service creates a new item in a document and optionally sets the item value. If an item of the same name already exists in a document, `appendItemValue` creates a second item of the same name, and the duplicate items are not accessible through most methods. If you are creating a new document, `appendItemValue` is safe. Otherwise `replaceltemValue` should be used. The parameter value is optional. The value may be the of type Integer Object, Double object or any Object. The data type of the new item depends upon the data type of the value placed in it.

Input Parameters

<i>document</i>	Document The lotus.domino.Document object to append an item to.
<i>name</i>	String The name of the item.
<i>value</i>	String Optional. The value of the item.

wm.In.document:computeWithForm

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:copyAllItems

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:copyItem

This flow service copies an item into the current document and optionally assigns the copied item a new name.

Input Parameters

<i>document</i>	Document The lotus.domino.Document object to copy an item to.
-----------------	--

<i>name</i>	Item The lotus.domino.Item representing the item to copy. Cannot be null.
<i>newName</i>	String Optional. The name to assign to the copied item. Specify null to retain the item's existing name.

Output Parameters

<i>newitem</i>	Item A lotus.domino.Item object representing the new Item.
----------------	---

wm.In.document:copyToDatabase

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:createReplyMessage

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:createRichTextItem

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:encrypt

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:generateXML

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getAttachment

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getAuthors

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getColumnValues

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getCreatedDateTime

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getEmbeddedObjects

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getEncryptionKeys

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getFirstItem

This Java service returns the first item of a specified name in a document.

Input Parameters

<i>document</i>	Document The lotus.domino.Document object from which to fetch the item.
<i>itemName</i>	String The name of the item to fetch.

Output Parameters

<i>item</i>	Item The item fetched.
<i>richtextItem</i>	RichTextItem The item fetched as a lotus.domino.RichTextItem

wm.In.document:getFolderReferences

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getItems

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getItemValue

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getLastAccessedDateTime

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getLastModifiedDateTime

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getParentDatabase

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getParentView

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:getProperties

This Java service is used to get the properties of the Document object.

Input Parameters

<i>document</i>	Document The lotus.domino.Document object to get the properties from.
-----------------	--

Output Parameters

<i>properties</i>	Record A record containing the following properties: <ul style="list-style-type: none">■ notelID■ FTSearchScore■ key■ nameOfProfile■ parentDocumentUNID■ signer■ size■ universalID■ verifier
-------------------	---

- hasEmbedded
- isEncryptOnSend
- isNewNote
- isProfile isResponse
- isSigned
- isValid
- isSaveMessageOnSend
- isSendByAgent
- isDeleted
- URL

wm.In.document:getResponses

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:hasItem

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:makeResponse

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:putInFolder

This flow service adds a document to a folder. If the folder does not exist, it is created.

Input Parameters

<i>document</i>	Document The document to add.
<i>folderName</i>	String The name of the folder in which to place the document. The folder may be personal if the program is running on a workstation. If the folder is within another folder, the path must be specified, separating folder names with backslashes, for example: "Vehicles \ Bikes"

createOnFail **String** Optional. If true (default), creates the folder if it does not exist.

wm.In.document:remove

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:removeFromFolder

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:removeItem

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:renderToRTItem

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:replaceItemValue

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:save

This flow service saves any changes made to a document.

Input Parameters

<i>document</i>	Document The lotus.domino.DateRange object.
<i>force</i>	String Optional. If true, the document is saved even if someone else edits and saves the document while the program is running. The last version of the document that was saved is kept; the earlier version is discarded. If false, and someone else edits the document while the program is running, the <i>makeresponse</i> parameter determines what happens.
<i>makeresponse</i>	String Optional. Requires the <i>force</i> parameter. If true, the current document becomes a response to the original document (this is what the replicator does when there's a replication conflict). If false, the save

is canceled. If the *force* parameter is true, the *makeresponse* parameter has no effect.

markread

String Optional. Requires all other parameters. If true, the document is marked as read. If false (default), the document is marked as unread.

Output Parameters

successFlag

String Indicates if the save operation completed.

wm.In.document:send

This flow service mails a document to the recipients you specify.

Input Parameters

document

Document The document to send.

recipients

String[] The recipients of the document, which may include people, groups, or mail-in databases. This list is ignored if the document contains a SendTo item, in which case the document is mailed to recipients listed in SendTo item. This list is required if the document does not contain a SendTo item.

attachform

String Optional. If true, the form is stored and sent along with the document. If false, it is not.

wm.In.document:setEncryptionKeys

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.document:setProperties

This Java service sets one or more properties on a document.

Input Parameters

document

Document The document object to set properties on.

properties

Record A record containing one or more of the following properties:

- universalID
 - encryptOnSend
 - saveMessageOnSend
 - signOnSend
-

wm.In.document:sign

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.documentCollection:addDocument

This flow service adds a document to a collection.

Input Parameters

<i>documentCollection</i>	DocumentCollection The document collection to add a document to.
<i>document</i>	Document The document to add. Cannot be null.
<i>checkDups</i>	String Optional. If true, forces a remote (IIOP) add to be made immediately rather than on the next navigation or other method (such as stampAll) that calls the server, so that a duplicate exception can be thrown immediately.

wm.In.documentCollection:deleteDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.documentCollection:FTSearch

This flow service conducts a full-text search of all the documents in a database collection, and reduces the collection to a sorted collection of those documents that match.

Input Parameters

<i>documentCollection</i>	DocumentCollection The document collection to search.
<i>query</i>	String The full-text query.

*maxDocs***String** The maximum number of documents you want returned from the query. Set this parameter to 0 to receive all matching documents.

wm.In.documentCollection:getCount

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.documentCollection:getFirstDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.documentCollection:getLastDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.documentCollection:getNextDocument

This flow service gets the document immediately following the current document or a specified document in a collection.

Input Parameters

*documentCollection***DocumentCollection** The document collection to get the document from.*document***Document** Optional. Any document in the collection. If input document = null, it returns the document following the current document, else the document following the specified document.

Output Parameters

*nextDocument***Document** The document fetched.

wm.In.documentCollection:getNthDocument

This flow service, given a position number, returns the document at that position in a collection. The preferred loop structure is getLastDocument() followed by getPrevDocument() until it returns null. For performance reasons, avoid using getNthDocument(int n) and getPrevDocument(Document).

Input Parameters

<i>documentCollection</i>	DocumentCollection The document collection to get the document from.
---------------------------	---

<i>dateRange</i>	DateRange The lotus.domino.DateRange object.
------------------	---

Output Parameters

<i>startDateTime</i>	DateTime The start date of the provided DateRange object as a lotus.domino.DateTime object.
----------------------	--

wm.In.documentCollection:getPrevDocument

This flow service gets the document immediately before the current document or a specified document in a collection.

Input Parameters

<i>documentCollection</i>	DocumentCollection The document collection to get the document from.
---------------------------	---

<i>document</i>	Document Optional. Any document in the collection. If input document = null it returns the document before the current document, else the document before the specified document.
-----------------	--

Output Parameters

<i>nextDocument</i>	Document The document fetched.
---------------------	---------------------------------------

wm.In.documentCollection:getQuery

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.documentCollection:hasDocument

This Java service determines if a document is in a document collection.

Input Parameters

<i>documentCollection</i>	DocumentCollection The document collection in which to search for a document.
<i>document</i>	Document The document to check for in a document collection.

Output Parameters

<i>hasDocument</i>	String "True" if the document exists, "false" otherwise.
--------------------	---

wm.In.documentCollection:isSorted

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.documentCollection:putAllInFolder

This flow service adds all the documents in the collection to the specified folder. If the folder does not exist in the document's database, it is created.

Input Parameters

<i>documentCollection</i>	DocumentCollection The document collection to add to a folder.
<i>folderName</i>	String The name of the folder in which to place the documents. The folder may be personal if the script is running on the workstation. If the folder is within another folder, specify a path to it, separating folder names with backward slashes, for example, "Vehicles\Bikes".
<i>createOnFail</i>	String If true (default), creates the folder if it does not exist.

wm.In.documentCollection:recycle

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.documentCollection:removeAll

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.documentCollection:removeAllFromFolder

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.documentCollection:stampAll

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.embeddedObject:extractFile

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.embeddedObject:getClassName

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.embeddedObject:getFileSize

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.embeddedObject:getInputStream

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.embeddedObject:getName

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.embeddedObject:getReader

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.embeddedObject:getSource

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.embeddedObject:getType

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.embeddedObject:remove

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.form:getAliases

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.form:getFields

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.form:getFormProperties and wm.In.form:getProperties

These Java services get the properties of a form.

Input Parameters

form **Form** The form object to get the properties for.

Output Parameters

properties **Record** A record containing the following properties:

- isProtectReaders
- isProtectUsers
- isSubForm
- URL

wm.In.form:getFormUsers

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.form:getParent

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.form:getReaders

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.form:recycle

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.form:remove

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.form:setFormProperties

This Java service sets one or more form properties.

Input Parameters

<i>form</i>	Form The form object to set the properties on.
<i>properties</i>	Record A record containing one or more of the following properties: <ul style="list-style-type: none">■ protectReaders■ protectUsers

wm.In.form:setFormUsers

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.form:setReaders

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.international:getProperties

This Java service gets the properties of the International object.

Input Parameters

<i>international</i>	International The international object to get the properties from.
----------------------	---

Output Parameters

<i>properties</i>	Record A record object containing the following properties: <ul style="list-style-type: none">■ AMString■ CurrencyDigits■ CurrencySymbol
-------------------	---

- DateSep
 - DecimalSep
 - PMString
 - ThousandsSep
 - TimeSep
 - TimeZone
 - Today
 - Tomorrow
 - Yesterday
 - isCurrencySuffix
 - isCurrencySpace
 - isCurrencyZero
 - isDateMDY
 - isDateDMY
 - isDateYMD
 - isDST
 - isTime24Hour
-

wm.In.item:addNewLine

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:addPageBreak

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:addTab

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:appendDbLink

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:appendDocLink

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:appendParagraphStyle

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:appendRTItem

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:appendStyle

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:appendText

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:appendViewLink

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:embedObject

This flow service, given the name of a file or an application, does one of the following:

- Attaches the file you specify to a rich-text item.
- Embeds an object in a rich-text item. The object is created using either the application or the file you specify.
- Places an object link in a rich-text item. The link is created using the file you specify.

embedObject is not supported under OS/2, under UNIX, and on the Macintosh.

Input Parameters

<i>item</i>	Item The item.
<i>type</i>	String One of the following values: <ul style="list-style-type: none">■ EMBED_ATTACHMENT■ EMBED_OBJECT

■ EMBED_OBJECTLINK

<i>class</i>	<p>Item Optional. If you are using EMBED_OBJECT and want to create an empty embedded object from an application, use this parameter to specify the name of the application (for example, "123Worksheet") and specify null for source. Case sensitive.</p> <p>If you are using EMBED_OBJECTLINK or EMBED_ATTACHMENT, specify an empty string ("").</p>
<i>source</i>	<p>String If you are using EMBED_OBJECT and want to create an embedded object from a file, use this parameter to specify the name of the file, and specify null for class. If you are using EMBED_ATTACHMENT or EMBED_OBJECTLINK, use this parameter to specify the name of the file to attach or link.</p>
<i>name</i>	<p>String Name by which you can reference the EmbeddedObject later. Can be null.</p>

Output Parameters

<i>embeddedObject</i>	<p>EmbeddedObject The newly attached file, embedded object, or linked object.</p> <p>Files can be attached on any supported platform, but objects and links can only be created on platforms supporting OLE. Files can be embedded as OLE/2 objects only on platforms supporting OLE, but they can be embedded as OLE/1 objects on any supported platform (if the file is of a supported type, such as a Lotus® application data file).</p>
-----------------------	--

wm.In.item:getDatetimeValue

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getEmbeddedObject

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getEmbeddedObjects

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getFormattedText

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getInputStream

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getLastModified

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getMIMEEntity

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getName

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getParentDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getProperties

This Java service gets the properties of an Item object.

Input Parameters

<i>item</i>	Item The Item object to get the properties for.
-------------	--

Output Parameters

<i>properties</i>	Record A record containing the following properties: <ul style="list-style-type: none">■ isAuthors■ isEncrypted■ isReaders■ isNames
-------------------	---

- isProtected
- isSaveToDisk
- isSigned
- isSummary

wm.In.item:getReader

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getText

This flow service gets a plain text representation of an item's value.

Input Parameters

<i>item</i>	Item The Item object to represent as plain text.
<i>maxlen</i>	String Optional. The maximum length of the returned text.

Output Parameters

<i>text</i>	String The text representation of the Item.
-------------	--

wm.In.item:getType

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getValue

This Java service returns the value of the item.

Input Parameters

<i>item</i>	Item The item whose value is to be retrieved.
<i>valueType</i>	String "String", "Integer" or "Double".

Output Parameters

<i>value</i>	String The value of the item rendered as a string from its initial type.
--------------	---

wm.In.item:getValueLength

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:getValues

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:remove

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:setDateTimeValue

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.item:setProperties

This Java service sets one or more properties of an Item object.

Input Parameters

<i>item</i>	Item The Item object on which to set properties.
<i>properties</i>	Record A record containing one or more of the following properties: <ul style="list-style-type: none">■ item■ isAuthors■ isEncrypted■ isReaders■ isNames■ isProtected■ isSaveToDisk

- isSigned
- isSummary

wm.In.item:setValue

This Java service sets the value of an Item object.

Input Parameters

<i>item</i>	Item The value of the provided item will be set.
<i>value</i>	String A string representation of the value of the Item.
<i>valueType</i>	String "String", "Integer" or "Double".

wm.In.item:setValues

This Java service sets the value of an item.

Input Parameters

<i>item</i>	DateRange The value of the Item will be set.
<i>value</i>	Object[] The data type of the value depends upon the type of the item.

wm.In.mimeEntity:getContentAsText

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.mimeEntity:getContentSubType

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.mimeEntity:getContentType

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.mimeEntity:getFirstChildEntity

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.mimeEntity:getHeaders

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.mimeEntity:getInputSource

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.mimeEntity:getInputStream

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.mimeEntity:getNextSibling

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.mimeEntity:getParentEntity

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.mimeEntity:getReader

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.name:getProperties

This Java service gets the properties of a Name object.

Input Parameters

<i>name</i>	Name The Name object to get the properties from.
-------------	---

Output Parameters

<i>properties</i>	Record A record containing the following properties: <ul style="list-style-type: none">■ ARMD■ Abbreviated■ Surname■ Initials
-------------------	---

- Common
 - Given
 - Country
 - Canonical
 - Generation
 - Addr822LocalPart
 - Addr822Phrase
 - Addr821
 - Addr822Comment1
 - Addr822Comment2
 - Addr822Comment3
 - Language
 - Keyword
 - Organization
 - OrgUnit1
 - OrgUnit2
 - OrgUnit3
 - OrgUnit4
 - PRMD
 - isHierarchical
-

wm.In.newsletter:createNewsletter

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.newsletter:formatDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.newsletter:formatMsgWithDoclinks

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.newsletter:getSubjectItemName

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.newsletter:isDoScore

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.newsletter:isDoSubject

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.newsletter:setDoScore

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.newsletter:setDoSubject

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.newsletter:setSubjectItemName

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.replication:getProperties

This Java service fetches the replication properties for the given database object.

Input Parameters

<i>replication</i>	Replication The Replication object to fetch properties for.
--------------------	--

Output Parameters

<i>properties</i>	Record A record containing the following properties:
-------------------	---

- Abstract
- CutOffDelete
- CutOffInterval
- CutOffDate

- isDisabled
- isIgnoreDeletes
- isIgnoreDestDeletes
- Priority

wm.In.replication:setProperties

This Java service sets one or more properties on a Replication object.

Input Parameters

<i>replication</i>	Replication The Replication object to set properties on.
<i>properties</i>	Record A record containing one or more of the following properties: <ul style="list-style-type: none"> ■ Abstract ■ CutOffDelete ■ CutOffInterval ■ isDisabled ■ isIgnoreDeletes ■ isIgnoreDestDeletes ■ Priority

wm.In.richText.richTextItem:addNewLine

This flow service appends one or more new lines (carriage returns) to the end of a rich-text item.

Input Parameters

<i>richTextItem</i>	RichTextItem The RichTextItem to add a new line to.
<i>count</i>	String Optional. The number of new lines to append.
<i>newparagraph</i>	String Optional. Must include all other parameters if using this parameter. If true (default), forces the new line to be a paragraph separator. If false, the new line is added.

wm.In.richText.richTextItem:addPageBreak

This flow service appends a page break to the end of a rich-text item.

Input Parameters

<i>richTextItem</i>	RichTextItem The RichTextItem to add a page break to.
<i>richTextParagraphStyle</i>	RichTextParagraphStyle Optional. The style to use.

wm.In.richText.richTextItem:addTab

This flow service adds one or more tabs to a rich text item.

Input Parameters

<i>richTextItem</i>	RichTextItem The RichTextItem to add tabs to.
<i>count</i>	String Optional. To add more than 1 tab, specify 2 or more.

wm.In.richText.richTextItem:appendDBLink

This flow service adds a doclink to the end of a rich-text item to the database to be linked.

Input Parameters

<i>richTextItem</i>	RichTextItem The RichTextItem to append a DocLink to.
<i>database</i>	Database The database to link to.
<i>comment</i>	String Optional. The text that appears when a user presses and holds the mouse over the link.
<i>hotspottext</i>	String Optional. If you use this parameter, you must also specify the comment parameter. Boxed text which the user clicks to follow the link. This text appears in place of a token.

wm.In.richText.richTextItem:appendDocLink

This flow service adds a doclink to the end of a rich-text item to the document to be linked.

Input Parameters

<i>richTextItem</i>	RichTextItem The RichTextItem to append a DocLink to.
<i>document</i>	Document The document to link to.
<i>comment</i>	String Optional. The text that appears when a user presses and holds the mouse over the link.
<i>hotspottext</i>	String Optional. If you use this parameter, you must also specify the comment parameter. Boxed text which the user clicks to follow the link. This text appears in place of a token.

wm.In.richText.richTextItem:appendParagraphStyle

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.richText.richTextItem:appendRTItem

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.richText.richTextItem:appendStyle

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.richText.richTextItem:appendText

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.richText.richTextItem:appendViewLink

This flow service adds a doclink to the end of a rich-text item to the View to be linked.

Input Parameters

<i>richTextItem</i>	RichTextItem The RichTextItem to append a View to.
<i>view</i>	View The view to link to.
<i>comment</i>	String Optional. The text that appears when a user presses and holds the mouse over the link.

hotspottext

String Optional. If you use this parameter, you must also specify the comment parameter. Boxed text which the user clicks to follow the link. This text appears in place of a token.

wm.In.richText.richTextItem:embedObject

This flow service, given the name of a file or an application, does one of the following:

- Attaches the file you specify to a rich-text item.
- Embeds an object in a rich-text item. The object is created using either the application or the file you specify.
- Places an object link in a rich-text item. The link is created using the file you specify.

embedObject is not supported under OS/2, under UNIX, and on the Macintosh.

Input Parameters

<i>item</i>	Item The item.
<i>type</i>	String One of the following values: <ul style="list-style-type: none">■ EMBED_ATTACHMENT■ EMBED_OBJECT■ EMBED_OBJECTLINK
<i>class</i>	Item Optional. If you are using EMBED_OBJECT and want to create an empty embedded object from an application, use this parameter to specify the name of the application (for example, "123Worksheet") and specify null for source. Case sensitive. If you are using EMBED_OBJECTLINK or EMBED_ATTACHMENT, specify an empty string ("").
<i>source</i>	String If you are using EMBED_OBJECT and want to create an embedded object from a file, use this parameter to specify the name of the file, and specify null for class. If you are using EMBED_ATTACHMENT or EMBED_OBJECTLINK, use this parameter to specify the name of the file to attach or link.

<i>name</i>	String Name by which you can reference the EmbeddedObject later. Can be null.
-------------	--

Output Parameters

<i>embeddedObject</i>	EmbeddedObject The newly attached file, embedded object, or linked object.
-----------------------	---

Files can be attached on any supported platform, but objects and links can only be created on platforms supporting OLE. Files can be embedded as OLE/2 objects only on platforms supporting OLE, but they can be embedded as OLE/1 objects on any supported platform (if the file is of a supported type, such as a Lotus® application data file).

wm.In.richText.richTextItem:getEmbeddedObject

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.richText.richTextItem:getEmbeddedObjects

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.richText.richTextItem:getFormattedText

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.richText.richTextStyle:getProperties

This Java service gets the properties of a RichTextStyle object.

Input Parameters

<i>richTextStyle</i>	RichTextStyle The RichTextStyle object to fetch properties for.
----------------------	--

Output Parameters

<i>properties</i>	Record A record containing the following properties:
-------------------	---

- Bold

- Effects
 - Color
 - Font
 - FontSize
 - Italic
 - PassThruHTML
 - StrikeThrough
 - Underline
-

wm.In.richText.richTextStyle:setProperties

This Java service sets one or more properties in a RichTextStyle object.

Input Parameters

richTextStyle

RichTextStyle The RichTextStyle object to set properties in.

Output Parameters

properties

Record A record containing one or more of following properties to set:

- richTextStyle
 - Bold
 - Effects
 - Color
 - Font
 - FontSize
 - Italic
 - PassThruHTML
 - StrikeThrough
 - Underline
-

- LeftMargin
 - Pagination
 - RightMargin
 - SpacingAbove
 - SpacingBelow
-

wm.In.richText.rtParagraphStyle:getTabs

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.richText.rtParagraphStyle:setProperties

This Java service sets one or more properties for the RichTextParagraphStyle object.

Input Parameters

richTextParagraphStyle

RichTextParagraphStyle The RichTextParagraphStyle object to set properties for.

Output Parameters

properties

Record A record containing one or more of the following properties:

- Alignment
 - FirstLineLeftMargin
 - InterLineSpacing
 - LeftMargin
 - Pagination
 - RightMargin
 - SpacingAbove
 - SpacingBelow
-

wm.In.richText.rtParagraphStyle:setTab

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.richText.rtParagraphStyle:setTabs

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.session:createNewsletter

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.session:createRichTextParagraphStyle

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.session:createRichTextStyle

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.session:evaluate

This flow service evaluates a Domino formula. If the formula contains the name of a field, you must use the second method parameter *document*. The formula takes the field from the document specified as parameter *document*. Functions that affect the user interface do not work in evaluate. These include: Command, DbManager, DbName, DbTitle, DDEExecute, DDEInitiate, DDEPoke, DDETerminate, DialogBox, PickList, PostedCommand, Prompt, and ViewTitle. You cannot change a document with evaluate; you can only get a result. To change a document, write the result to the document with a method such as Document.replaceItemValue.

Input Parameters

<i>session</i>	Session The Session object to evaluate the formula with.
<i>formula</i>	String The Domino formula.
<i>document</i>	Document Optional. The formula context. Cannot be null.

Output Parameters

<i>value</i>	String[] The result of the evaluation. A scalar result is returned in firstElement.
--------------	---

wm.In.session:freeTimeSearch

This flow service searches for free time slots for calendaring and scheduling.

Input Parameters

<i>session</i>	Session The Session object to search for free time.
<i>DateRange</i>	DateRange Starting and ending times within which to search for a match. Cannot be null.
<i>duration</i>	String The length in minutes of the free-time interval you want to schedule.
<i>names</i>	String[] The name or names of the people or groups whose free time you want to schedule. Cannot be null.
<i>firstfit</i>	String "True" if you just want the first matching date range. "False" if you want all matching times.

Output Parameters

<i>availDateRanges</i>	DateRange[] Available times slots. Null if there are no matches.
<i>freeTimeFound</i>	String "True" if one or more slots are found, "False" otherwise.

wm.In.session:getAddressBooks

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.session:getDatabase

This flow service creates a database object that represents a database located at the server and file name you specify, and opens the database, if possible.

Input Parameters

<i>session</i>	Session The Session object to open the database with.
<i>dbName</i>	String The file name and location of the database within the Domino data directory. Use a full path name if the database is not within the Domino data directory.
<i>createOnFail</i>	String Optional. If true, creates a Database object even if the specified database cannot be opened. If false or omitted, returns null if the database cannot be opened.

Output Parameters

<i>database</i>	Database A Database object that can be used to access the database specified.
-----------------	--

wm.In.session:getDbDirectory

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.session:getEnvironmentString

This flow service fetches the environment variable for the given session object. To access System environment variable set *issystem* to "true". For user defined variable set it to "false" the default is "true".

Input Parameters

<i>session</i>	Session The session object to fetch a variable with.
<i>vname</i>	String The name of the variable.
<i>issystem</i>	String Optional. If "false", the variable is a user defined variable, if "true" it is a system variable.

Output Parameters

<i>envString</i>	String The value of the environment variable.
------------------	--

wm.In.session:getInternational

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.session:getInternationalProperties

This flow service is equivalent to calling `wm.In.session:getInternational` followed by `wm.In.international:getProperties`.

wm.In.session:getProperties

This Java service gets the properties of a session object.

Input Parameters

session

Session The Session object to get properties for.

Output Parameters

properties

Record A record containing the following properties:

- Platform
 - NotesVersion
 - IsOnServer
 - UserName
-

wm.In.session:getURLDatabase

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.session:getUserNameObject

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.session:openDatabase

This flow service is equivalent to calling `wm.In.session:getDbDirectory` followed by `wm.In.dbDirectory:openDatabase`.

wm.In.session:resolve

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.session:setEnvironmentString

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.utils.garbage:recycle

This Java service recycles any Domino Object in the pipeline including Session objects. All other objects are recycled first. Use this service while ensuring that no Session object is in the pipeline to recycle objects that you no longer intend to use, and wish to keep a Session object you created. You may also use this service in case you run into a failure and want to ensure that any objects

created up to the failure point are recycled. For more information about when to use this service in a flow, see [“Freeing Resources After Processing” on page 16](#).

Input Parameters

<i>envString</i>	String The value of the environment variable.
------------------	--

wm.In.view:clear

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:createViewNav

This flow service creates a view navigator for all entries in a view.

Input Parameters

<i>view</i>	View The View object to create a ViewNavigator object for.
-------------	---

<i>cacheSize</i>	String Optional. The size of the navigator cache in view entries. Legal values are 0 (no cache) through 128 (default).
------------------	---

Output Parameters

<i>viewNavigator</i>	ViewNavigator The created ViewNavigator object for the View object provided.
----------------------	---

wm.In.view:createViewNavFromCategory

This flow service is a wrapper for a lotus.domino.class and method. It accepts an optional *cacheSize* parameter in addition to the standard parameters of the method being wrapped.

wm.In.view:createViewNavFromChildren

This flow service is a wrapper for a lotus.domino.class and method. It accepts an optional *cacheSize* parameter in addition to the standard parameters of the method being wrapped.

wm.In.view:createViewNavFromDescendants

This flow service is a wrapper for a lotus.domino.class and method. It accepts an optional *cacheSize* parameter in addition to the standard parameters of the method being wrapped.

wm.In.view:createViewNavFromDocument

This flow service is a wrapper for a `lotus.domino.class` and method. It accepts an optional `cacheSize` parameter in addition to the standard parameters of the method being wrapped.

wm.In.view:createViewNavFromViewEntry

This flow service is a wrapper for a `lotus.domino.class` and method. It accepts an optional `cacheSize` parameter in addition to the standard parameters of the method being wrapped.

wm.In.view:createViewNavMaxLevel

This flow service is a wrapper for a `lotus.domino.class` and method. It accepts an optional `cacheSize` parameter in addition to the standard parameters of the method being wrapped.

wm.In.view:FTSearch

This flow service conducts a full-text search of all the documents in a database collection, and reduces the collection to a sorted collection of those documents that match.

Input Parameters

<i>view</i>	View The View object to conduct the search with.
<i>vname</i>	String The full-text query.
<i>maxDocs</i>	String Optional. The maximum number of documents you want returned from the query. Set this parameter to 0 to receive all matching documents.

Output Parameters

<i>numberOfResults</i>	String The output of the search execution.
------------------------	---

wm.In.view:getAliases

This Java service is a wrapper for a `lotus.domino.class` and method.

wm.In.view:getAllDocumentsByKey

This Java service finds documents based on their column values within a view. You create an array of keys, where each key corresponds to a value in a sorted column in the view. The method returns all documents whose column values match the keys.

Input Parameters

<i>view</i>	View The View object to get documents from.
<i>key</i>	Object[] Each key may be a String, Number, DateTime, or DateRange objects that are compared to sorted columns in the view. The first element in the vector is compared to the first sorted column in the view; the second element is compared to the second sorted column; and so on. Or it may be A single String, Number, DateTime, or DateRange object that is compared to the first sorted column in the view.
<i>Exact</i>	String Specify "true" if you want to find an exact match. If you specify "false" or omit this parameter, a partial match succeeds.

Output Parameters

<i>documentCollection</i>	DocumentCollection All documents in the view whose column values match each of the keys. If no documents match, the collection is empty and the count is zero.
---------------------------	---

wm.In.view:getAllEntries

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getAllEntriesByKey

This Java service finds view entries of type Document based on their column values within a view. You create a key or vector of keys, where each key corresponds to a value in a sorted column in the view. The method returns all entries whose column values match the keys.

Input Parameters

<i>view</i>	View The view to fetch entries from.
<i>key</i>	Object[] Each key may be either a Vector or Object of type String, DateTime, or DateRange. A key may also be a number.
<i>exact</i>	String Optional. "true" if you want to find an exact match. If you specify "false" or omit this parameter, a partial match succeeds.

Output Parameters

<i>viewEntryCollection</i>	ViewEntryCollection A collection of view entries found.
----------------------------	--

wm.In.view:getChildDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getColumn

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getColumnNames

This Java service is a wrapper for a lotus.domino.class and method.

wm.In.view:getColumns

This Java service is a wrapper for a lotus.domino.class and method.

wm.In.view:getDocumentByKey

This Java service finds a document based on its column values within a view. You create a key or vector of keys, where each key corresponds to a value in a sorted column in the view. The method returns the first document with column values that match the keys.

Input Parameters

<i>view</i>	View The View object to fetch documents from.
<i>key</i>	String Each key may be either a Vector or Object of type String, DateTime, or DateRange. A key may also be a number.
<i>exact</i>	String Optional. "true" if you want to find an exact match. If you specify "false" or omit this parameter, a partial match succeeds.

Output Parameters

<i>document</i>	Document The Document object fetched.
-----------------	--

wm.In.view:getEntryByKey

This Java service finds a view entry based on its column values within a view. You create a key or vector of keys, where each key corresponds to a value in a sorted column in the view. The method returns the first entry with column values that match the keys.

Input Parameters

<i>view</i>	View The View object to fetch entries from.
<i>key</i>	DateRange, DateTime, Number, Vector, or String Each key may be either a Vector or Object of type String, DateTime, or DateRange. A key may also be a number.
<i>exact</i>	String Optional. "true" if you want to find an exact match. If you specify "false" or omit this parameter, a partial match succeeds.

Output Parameters

<i>viewEntry</i>	ViewEntry The Document object fetched.
------------------	---

wm.In.view:getFirstDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getLastDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getNextDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getNextSibling

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getNthDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getParentDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getPrevDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getPrevSibling

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:getProperties

This Java service gets properties of a View object.

Input Parameters

view

View The View object to get properties from.

Output Parameters

properties

Record A record containing the following properties:

- ColumnCount
- Roines
- Spacing
- TopLevelEntryCount
- UniversalId
- isAutoUpdate
- isCalender
- isCategorized
- isConflict
- isDefaultView
- isFolder
- isHierarchical

- BackgroundColor
- HeaderLines
- isModified
- isPrivate
- isProtectReaders

wm.In.view:getProperty

This Java service gets a certain property of a View object.

Input Parameters

<i>view</i>	View The View object to get a property from.
<i>property</i>	String The name of the property to fetch.

Output Parameters

<i>value</i>	Object The value of the property fetched.
--------------	--

wm.In.view:getReaders

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:refresh

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:remove

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:setAutoUpdate

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:setProtectReaders

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.view:setReaders

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewColumn:getParentView

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewColumn:getProperties

This Java service gets properties of a ViewColumn object.

Input Parameters

viewColumn

ViewColumn The ViewColumn object to get properties from.

Output Parameters

properties

Record A record containing the following properties:

- `fontFace`
- `fontStyle`
- `fontColor`
- `fontPointSize`
- `numberDigits`
- `numberFormat`
- `numberAttrib`
- `dateFmt`
- `timeFmt`
- `timeZoneFmt`
- `timeDateFmt`
- `position`
- `formula`
- `width`

- listStep
- alignment
- headerAlignment

wm.In.viewColumn:getProperty

This Java service gets a certain property of a ViewColumn object.

Input Parameters

<i>viewColumn</i>	ViewColumn The ViewColumn object to get a property from.
<i>property</i>	String The name of the property to fetch.

Output Parameters

<i>value</i>	Object The value of the property fetched.
--------------	--

wm.In.viewEntry:getColumnValues

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntry:getDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntry:getParent

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntry:getProperties

This Java service gets properties of a ViewEntry object.

Input Parameters

<i>viewEntry</i>	ViewEntry The ViewEntry object to get properties from.
------------------	---

Output Parameters

properties

Record A record containing the following properties:

- isDocument
 - isCategory
 - isTotal
 - SiblingCount
 - ChildCount
 - DescendentCount
 - IndentLevel
 - ColumnIndentLevel
 - Noteid
 - UniversalID
 - isConflict
 - FTSearchScore
 - isValid
-

wm.In.viewEntry:getProperty

This Java service gets a certain property of a ViewColumn object.

Input Parameters

viewEntry

ViewEntry The ViewEntry object to get a property from.

property

String The name of the property to fetch.

Output Parameters

value

Object The value of the property fetched.

wm.In.viewEntryCollection:addEntry

This flow service adds an entry to a ViewEntryCollection.

Input Parameters

<i>viewEntryCollection</i>	ViewEntryCollection The ViewEntryCollection object to add an entry to.
<i>object</i>	Object A ViewEntry or Document object representing the entry to be added. Cannot be null. A ViewEntry object must be of type document.
<i>checkDups</i>	String Optional. If "true", forces a remote (IIOP) add to be made immediately rather than at the next navigation or other method (such as stampAll) that calls the server, so that a duplicate exception can be thrown immediately.

wm.In.viewEntryCollection:deleteEntry

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntryCollection:FTSearch

This flow service conducts a full-text search on all entries in a view entry collection and filters the collection so that it represents only the entries that match the full-text query.

Input Parameters

<i>viewEntryCollection</i>	ViewEntryCollection The ViewEntryCollection object to search.
<i>query</i>	String The full-text query.
<i>maxDocs</i>	String The maximum number of documents you want returned from the query. Set this parameter to 0 to receive all matching documents.

wm.In.viewEntryCollection:getCount

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntryCollection:getEntry

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntryCollection:getFirstEntry

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntryCollection:getLastEntry

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntryCollection:getNextEntry

This flow service gets the entry immediately following the current or a specified entry in a view entry collection.

Input Parameters

<i>viewEntryCollection</i>	ViewEntryCollection The ViewEntryCollection object to get the next entry from.
<i>viewEntry</i>	ViewEntry Optional. Any entry in the view entry collection. Cannot be null.

Output Parameters

<i>nextViewEntry</i>	ViewEntry The entry following the current or specified entry. If there is no next entry, returns null.
----------------------	---

wm.In.viewEntryCollection:getNthEntry

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntryCollection:getParent

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntryCollection:getPrevEntry

This flow service gets the entry immediately before the current or a specified entry in a view entry collection.

Input Parameters

<i>viewEntryCollection</i>	ViewEntryCollection The ViewEntryCollection object to get the previous entry from.
<i>viewEntry</i>	ViewEntry Optional. Any entry in the view entry collection. Cannot be null.

Output Parameters

<i>prevViewEntry</i>	ViewEntry The entry before the current or specified entry. If there is no next entry, returns null.
----------------------	--

wm.In.viewEntryCollection:getQuery

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntryCollection:putAllInFolder

This flow service adds the documents associated with all the entries in a view entry collection to a specified folder.

Input Parameters

<i>viewEntryCollection</i>	ViewEntryCollection The ViewEntryCollection object to use to add documents to a specified folder.
<i>folderName</i>	String The name of the folder in which to place the documents. The folder may be personal if the program is running on the workstation. If the folder is within another folder, specify a path to it, separating folder names with backslashes and escaping the backslashes in a literal, for example: <code>viewEntryCollection.putAllInFolder("Vehicles \ \ Bikes");</code>
<i>createOnFail</i>	String Optional. If "true" (default), creates the folder if it does not exist. If "false" and the folder does not exist, this method does nothing.

wm.In.viewEntryCollection:removeAll

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntryCollection:removeAllFromFolder

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewEntryCollection:stampAll

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getCacheSize

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getChild

This flow service returns the first child of an entry in a ViewNavigator. This method moves the current pointer to the retrieved entry unless the return value is null. A child of a category entry can be a category or document entry.

Input Parameters

<i>viewNavigator</i>	ViewNavigator The ViewNavigator object to get the first child entry from.
----------------------	--

<i>entry</i>	ViewEntry Optional. An entry in the view. Defaults to the current entry. Cannot be null.
--------------	---

Output Parameters

<i>childViewEntry</i>	ViewEntry The first child of the current or specified entry. Returns null if there are no children.
-----------------------	--

wm.In.viewNavigator:getCurrent

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getFirst

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getFirstDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getLast

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getLastDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getMaxLevel

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getNext

This flow service returns the next entry following the current or specified entry in a view navigator.

Input Parameters

<i>viewNavigator</i>	ViewNavigator The ViewNavigator object to get the next entry from.
<i>entry</i>	ViewEntry Optional. An entry in the view. Cannot be null.

Output Parameters

<i>nextViewEntry</i>	ViewEntry The entry following the current or specified entry. Returns null if there is no next entry.
----------------------	--

wm.In.viewNavigator:getNextCategory

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getNextDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getNthEntry

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getParent

This flow service returns the parent entry of the current or a specified entry in a view navigator. This method moves the current pointer to the retrieved entry unless the return value is null.

Input Parameters

<i>viewNavigator</i>	ViewNavigator The ViewNavigator object to get a parent entry with.
----------------------	---

entry **ViewEntry** Optional. An entry in the view. Cannot be null.

Output Parameters

parentViewEntry **ViewEntry** The parent entry of the current or specified entry. Returns null if there is no parent entry

wm.In.viewNavigator:getParentView

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getPos

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getPrev

This flow service returns the entry before the current or a specified entry in a view navigator. This method moves the current pointer to the retrieved entry unless the return value is null.

Input Parameters

viewNavigator **ViewNavigator** The ViewNavigator object to get a previous entry with.

viewEntry **ViewEntry** Optional. An entry in the view. Cannot be null.

Output Parameters

Properties **Record** A record containing the following properties:

wm.In.viewNavigator:getPrevCategory

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getPrevDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:getPrevSibling

This flow service returns the previous sibling entry to the current or specified entry in a view navigator.

Input Parameters

<i>viewNavigator</i>	ViewNavigator The ViewNavigator object to get the previous sibling from.
<i>viewEntry</i>	ViewEntry Optional. An entry in the view. Cannot be null.

Output Parameters

<i>prevSiblingViewEntry</i>	ViewEntry The sibling entry before the current or specified entry. Returns null if there is no previous sibling entry.
-----------------------------	---

wm.In.viewNavigator:gotoChild

This flow service moves the current pointer to the first child of the current or specified entry in a view navigator. A child of a category entry can be a category or document entry.

Input Parameters

<i>viewNavigator</i>	ViewNavigator The ViewNavigator object to use to point to a child entry.
<i>viewEntry</i>	ViewEntry Optional. An entry in the view. Cannot be null.

Output Parameters

<i>success</i>	Record "true" if the operation succeeds or "false" if there are no children.
----------------	---

wm.In.viewNavigator:gotoEntry

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:gotoFirst

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:gotoFirstDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:gotoLast

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:gotoLastDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:gotoNext

This flow service moves the current pointer to the next entry of the current or specified entry in a view navigator.

Input Parameters

<i>viewNavigator</i>	ViewNavigator The ViewNavigator object to move a pointer to an entry in.
<i>viewEntry</i>	ViewEntry Optional. An entry in the view. Cannot be null.

Output Parameters

<i>success</i>	Record "true" if the operation succeeds or "false" if there are no children.
----------------	---

wm.In.viewNavigator:gotoNextCategory

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:gotoNextDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:gotoNextSibling

This flow service moves the current pointer to the next sibling entry of the current or specified entry in a view navigator.

Input Parameters

<i>viewNavigator</i>	ViewNavigator The ViewNavigator object to use to go to an indicated entry.
----------------------	---

<i>viewEntry</i>	ViewEntry Optional. An entry in the view. Cannot be null.
------------------	--

Output Parameters

<i>success</i>	Record "true" if the operation succeeds or "false" if there are no children.
----------------	---

wm.In.viewNavigator:gotoParent

This flow service moves the current pointer to the parent entry of the current or specified entry in a view navigator.

Input Parameters

<i>viewNavigator</i>	ViewNavigator The ViewNavigator object to move the pointer in.
----------------------	---

<i>viewEntry</i>	ViewEntry Optional. An entry in the view. Cannot be null.
------------------	--

Output Parameters

<i>success</i>	Record "true" if the operation succeeds or "false" if there are no children.
----------------	---

wm.In.viewNavigator:gotoPos

This flow service gets properties of a View object.

Input Parameters

View **View** The View object to get properties from.

Output Parameters

Properties **Record** A record containing the following properties:

wm.In.viewNavigator:gotoPrev

This flow service moves the current pointer to the previous entry of the current or specified entry in a view navigator.

Input Parameters

viewNavigator **ViewNavigator** The ViewNavigator object to move a pointer to an entry in.

viewEntry **ViewEntry** Optional. An entry in the view. Cannot be null.

Output Parameters

success **Record** "true" if the operation succeeds or "false" if there are no children.

wm.In.viewNavigator:gotoPrevCategory

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:gotoPrevDocument

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:gotoPrevSibling

This flow service moves the current pointer to the previous sibling entry of the current or specified entry in a view navigator.

Input Parameters

<i>viewNavigator</i>	ViewNavigator The ViewNavigator object to use to go to an indicated entry.
<i>viewEntry</i>	ViewEntry Optional. An entry in the view. Cannot be null.

Output Parameters

<i>success</i>	Record "true" if the operation succeeds or "false" if there are no children.
----------------	---

wm.In.viewNavigator:setCacheSize

This flow service is a wrapper for a lotus.domino.class and method.

wm.In.viewNavigator:setMaxLevel

This flow service is a wrapper for a lotus.domino.class and method.

