



Workbench Guide

- SAP BusinessObjects Data Services 4.1 Support Package 1 (14.1.1.0)

2012-12-12

Copyright

© 2012 SAP AG. All rights reserved. SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP BusinessObjects Explorer, StreamWork, SAP HANA and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries. Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects Software Ltd. Business Objects is an SAP company. Sybase and Adaptive Server, iAnywhere, Sybase 365, SQL Anywhere, and other Sybase products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Sybase, Inc. Sybase is an SAP company. Crossgate, m@gic EDDY, B2B 360°, B2B 360° Services are registered trademarks of Crossgate AG in Germany and other countries. Crossgate is an SAP company. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary. These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

2012-12-12

Contents

Chapter 1	Introduction.....5
1.1	Welcome to SAP BusinessObjects Data Services.....5
1.1.1	Welcome.....5
1.1.2	Documentation set for SAP BusinessObjects Data Services.....5
1.1.3	Accessing documentation.....8
1.1.4	SAP BusinessObjects information resources.....9
1.2	Overview of this guide.....9
1.2.1	About this guide.....10
1.2.2	Who should read this guide.....10
1.3	Naming Conventions.....10
Chapter 2	Workbench User Interface.....15
2.1	Logging into the Workbench.....15
2.2	Workbench window.....15
2.3	Menu bar.....16
2.3.1	File menu.....16
2.3.2	Edit menu.....17
2.3.3	Tools menu.....17
2.3.4	Validate menu.....18
2.3.5	Window menu.....18
2.3.6	Help menu.....19
2.4	Toolbar.....19
2.5	Project explorer.....20
2.6	Replication job editor.....20
2.7	Datastore editor.....22
2.8	Monitoring editor.....23
Chapter 3	Database Migration.....25
3.1	Migrating data and schema information.....25
3.1.1	Creating a Workbench project.....26
3.1.2	Using the quick replication wizard.....26
3.1.3	Using the datastore editor.....28

3.1.4	Using the replication job editor.....	29
3.1.5	Deploying and executing the replication job.....	34
3.2	Delta load migration.....	36
3.2.1	Delta load jobs in the Workbench.....	36
3.2.2	Delta load options for tables.....	37
3.2.3	Delta load options for SAP applications.....	37
Chapter 4	Additional Information.....	39
4.1	Supported data migration sources and targets.....	39
4.2	Using the expression editor.....	40
4.2.1	Supported operators.....	40
4.2.2	Including functions in expressions.....	41
4.3	Managing Workbench project information.....	49
4.3.1	Creating a Workbench project.....	49
4.3.2	Exporting resources and preferences.....	50
4.3.3	Importing resources and preferences.....	51
Index		53

Introduction

1.1 Welcome to SAP BusinessObjects Data Services

1.1.1 Welcome

SAP BusinessObjects Data Services delivers a single enterprise-class solution for data integration, data quality, data profiling, and text data processing that allows you to integrate, transform, improve, and deliver trusted data to critical business processes. It provides one development UI, metadata repository, data connectivity layer, run-time environment, and management console—enabling IT organizations to lower total cost of ownership and accelerate time to value. With SAP BusinessObjects Data Services, IT organizations can maximize operational efficiency with a single solution to improve data quality and gain access to heterogeneous sources and applications.

1.1.2 Documentation set for SAP BusinessObjects Data Services

You should become familiar with all the pieces of documentation that relate to your SAP BusinessObjects Data Services product.

Document	What this document provides
<i>Administrator's Guide</i>	Information about administrative tasks such as monitoring, lifecycle management, security, and so on.
<i>Customer Issues Fixed</i>	Information about customer issues fixed in this release.
<i>Designer Guide</i>	Information about how to use SAP BusinessObjects Data Services Designer.
<i>Documentation Map</i>	Information about available SAP BusinessObjects Data Services books, languages, and locations.

Document	What this document provides
<i>Installation Guide for Windows</i>	Information about and procedures for installing SAP BusinessObjects Data Services in a Windows environment.
<i>Installation Guide for UNIX</i>	Information about and procedures for installing SAP BusinessObjects Data Services in a UNIX environment.
<i>Integrator's Guide</i>	Information for third-party developers to access SAP BusinessObjects Data Services functionality using web services and APIs.
<i>Master Guide</i>	Information about the application, its components and scenarios for planning and designing your system landscape. Information about SAP BusinessObjects Information Steward is also provided in this guide.
<i>Management Console Guide</i>	Information about how to use SAP BusinessObjects Data Services Administrator and SAP BusinessObjects Data Services Metadata Reports.
<i>Performance Optimization Guide</i>	Information about how to improve the performance of SAP BusinessObjects Data Services.
<i>Reference Guide</i>	Detailed reference material for SAP BusinessObjects Data Services Designer.
<i>Release Notes</i>	Important information you need before installing and deploying this version of SAP BusinessObjects Data Services.
<i>Technical Manuals</i>	<p>A compiled “master” PDF of core SAP BusinessObjects Data Services books containing a searchable master table of contents and index:</p> <ul style="list-style-type: none"> • <i>Administrator's Guide</i> • <i>Designer Guide</i> • <i>Reference Guide</i> • <i>Management Console Guide</i> • <i>Performance Optimization Guide</i> • <i>Supplement for J.D. Edwards</i> • <i>Supplement for Oracle Applications</i> • <i>Supplement for PeopleSoft</i> • <i>Supplement for Salesforce.com</i> • <i>Supplement for Siebel</i> • <i>Supplement for SAP</i> • <i>Workbench Guide</i>
<i>Text Data Processing Extraction Customization Guide</i>	Information about building dictionaries and extraction rules to create your own extraction patterns to use with Text Data Processing transforms.
<i>Text Data Processing Language Reference Guide</i>	Information about the linguistic analysis and extraction processing features that the Text Data Processing component provides, as well as a reference section for each language supported.
<i>Tutorial</i>	A step-by-step introduction to using SAP BusinessObjects Data Services.

Document	What this document provides
<i>Upgrade Guide</i>	Release-specific product behavior changes from earlier versions of SAP BusinessObjects Data Services to the latest release. This manual also contains information about how to migrate from SAP BusinessObjects Data Quality Management to SAP BusinessObjects Data Services.
<i>What's New</i>	Highlights of new key features in this SAP BusinessObjects Data Services release. This document is not updated for support package or patch releases.
<i>Workbench Guide</i>	Provides users with information about how to use the Workbench to migrate data and database schema information between different database systems.

In addition, you may need to refer to several Supplemental Guides.

Document	What this document provides
<i>Supplement for J.D. Edwards</i>	Information about interfaces between SAP BusinessObjects Data Services and J.D. Edwards World and J.D. Edwards OneWorld.
<i>Supplement for Oracle Applications</i>	Information about the interface between SAP BusinessObjects Data Services and Oracle Applications.
<i>Supplement for PeopleSoft</i>	Information about interfaces between SAP BusinessObjects Data Services and PeopleSoft.
<i>Supplement for Salesforce.com</i>	Information about how to install, configure, and use the SAP BusinessObjects Data Services Salesforce.com Adapter Interface.
<i>Supplement for SAP</i>	Information about interfaces between SAP BusinessObjects Data Services, SAP Applications, SAP Master Data Services, and SAP NetWeaver BW.
<i>Supplement for Siebel</i>	Information about the interface between SAP BusinessObjects Data Services and Siebel.

We also include these manuals for information about SAP BusinessObjects Information platform services.

Document	What this document provides
<i>Information Platform Services Administrator's Guide</i>	Information for administrators who are responsible for configuring, managing, and maintaining an Information platform services installation.
<i>Information Platform Services Installation Guide for UNIX</i>	Installation procedures for SAP BusinessObjects Information platform services on a UNIX environment.
<i>Information Platform Services Installation Guide for Windows</i>	Installation procedures for SAP BusinessObjects Information platform services on a Windows environment.

1.1.3 Accessing documentation

You can access the complete documentation set for SAP BusinessObjects Data Services in several places.

1.1.3.1 Accessing documentation on Windows

After you install SAP BusinessObjects Data Services, you can access the documentation from the Start menu.

1. Choose **Start > Programs > SAP BusinessObjects Data Services 4.1 > Data Services Documentation > All Guides**.
2. Click the appropriate shortcut for the document that you want to view.

1.1.3.2 Accessing documentation on UNIX

After you install SAP BusinessObjects Data Services, you can access the documentation by going to the directory where the printable PDF files were installed.

1. Go to `<LINK_DIR>/doc/book/en/`.
2. Using Adobe Reader, open the PDF file of the document that you want to view.

1.1.3.3 Accessing documentation from the Web

You can access the complete documentation set for SAP BusinessObjects Data Services from the SAP BusinessObjects Business Users Support site.

To do this, go to <http://help.sap.com/bods>.

You can view the PDFs online or save them to your computer.

1.1.4 SAP BusinessObjects information resources

A global network of SAP BusinessObjects technology experts provides customer support, education, and consulting to ensure maximum information management benefit to your business.

Useful addresses at a glance:

Address	Content
Customer Support, Consulting, and Education services http://service.sap.com/	Information about SAP Business User Support programs, as well as links to technical articles, downloads, and online forums. Consulting services can provide you with information about how SAP BusinessObjects can help maximize your information management investment. Education services can provide information about training options and modules. From traditional classroom learning to targeted e-learning seminars, SAP BusinessObjects can offer a training package to suit your learning needs and preferred learning style.
Product documentation http://help.sap.com/bods/	SAP BusinessObjects product documentation.
Supported Platforms (Product Availability Matrix) https://service.sap.com/PAM	Get information about supported platforms for SAP BusinessObjects Data Services. Use the search function to search for Data Services. Click the link for the version of Data Services you are searching for.

1.2 Overview of this guide

Welcome to the *Workbench Guide*. The Data ServicesWorkbench provides a graphical user interface (GUI) development environment in which you define data application logic to migrate data and database schema information between different databases in a data warehousing environment.

Key features of the Workbench include:

- Browsing table metadata and data.
- Selecting individual source tables for migration.

- Specifying the order in which the source tables should be migrated.
- Adjusting the table schema in detail. For example, adding or removing columns, defining constraints, partitions, indexes, and so on.
- Specifying filters and simple projection expressions.
- Specifying source and target table options such as array fetch size and bulk-loading options.
- Executing a replication job as an initial load or delta load.

1.2.1 About this guide

This guide contains two kinds of information:

- Conceptual information that helps you understand the Data ServicesWorkbench and how it works.
- Procedural information that explains in a step-by-step manner how to accomplish a task.

You will find this guide most useful:

- While you are learning about the product.
- While you are performing tasks in the design and early testing phases of your data-movement projects.
- As a general source of information during any phase of your projects.

1.2.2 Who should read this guide

This and other Data Services product documentation assumes the following:

- You are an application developer, consultant, or database administrator working on data warehousing.
- You understand your source data systems and RDMBS.
- You understand your organization's data needs.
- You are familiar with SQL (Structured Query Language).
- You are familiar with Data Services installation environments — Microsoft Windows or UNIX.

1.3 Naming Conventions

In this documentation, the following naming conventions apply:

Terminology

- “Data Services system” refers to “SAP BusinessObjects Data Services”.
- “BI platform” refers to “SAP BusinessObjects BI platform”.

Note:

The BI platform components required by Data Services may also be provided by SAP BusinessObjects Information platform services (IPS).

- “CMC” refers to the Central Management Console provided by the BI or IPS platform.
- “CMS” refers to the Central Management Server provided by BI or IPS platform.

Variables

Variables	Description
<INSTALL_DIR>	<p>The installation directory for the SAP BusinessObjects software.</p> <p>Default location:</p> <ul style="list-style-type: none"> Windows platforms C:\Program Files (x86)\SAP BusinessObjects UNIX platforms \$HOME/businessobjects
<BIP_INSTALL_DIR>	<p>The root directory of the BI or IPS platform.</p> <p>Default location:</p> <ul style="list-style-type: none"> Windows platforms <INSTALL_DIR>/SAP BusinessObjects Enterprise XI 4.0 UNIX platforms <INSTALL_DIR>/enterprise_xi40 <p>Note: These paths are the same for both the SAP BusinessObjects BI platform and SAP BusinessObjects Information platform services.</p>
<LINK_DIR>	<p>The root directory of the Data Services system.</p> <p>Default location:</p> <ul style="list-style-type: none"> All platforms <INSTALL_DIR>/Data Services <p>This system environment variable is created automatically during installation.</p>
<DS_COMMON_DIR>	

Variables	Description
	<p>The common configuration directory for the Data Services system.</p> <p>Default location:</p> <ul style="list-style-type: none"> Windows (Vista and newer) <code>%SYSTEMDRIVE%\ProgramData\SAP BusinessObjects\Data Services</code> Windows (Older versions) <code>%SYSTEMDRIVE%\Documents and Settings\All Users\Application Data\SAP BusinessObjects\Data Services</code> UNIX systems (for compatibility) <code><LINK_DIR></code> <p>This system environment variable is created automatically during installation.</p>
<code><DS_USER_DIR></code>	<p>The user-specific configuration directory for the Data Services system.</p> <p>Default location:</p> <ul style="list-style-type: none"> Windows (Vista and newer) <code>%SYSTEMDRIVE%\Users\username\AppData\Local\SAP BusinessObjects\Data Services</code> Windows (Older versions) <code>%SYSTEMDRIVE%\Documents and Settings\username\Application Data</code> <p>This user environment variable is created automatically during installation.</p> <p>Note: This variable is used only for Data Services client applications on Windows, such as the Designer. <code>DS_USER_DIR</code> is not used on UNIX platforms.</p>

Workbench User Interface

This section provides basic information about the Workbench user interface.

2.1 Logging into the Workbench

Before you can use the Workbench to create, edit, and execute replication jobs, you must log in as a user defined in the Central Management Server (CMS).

1. Launch the Workbench.

The Workbench can be accessed from the Windows Start Menu: **All Programs > SAP BusinessObjects Data Services 4.1 > Data Services Workbench**.

2. Enter your user credentials for the CMS.

Option	Description
System	The server name and optionally the port number for the CMS.
User name	The user name to use to log into the CMS.
Password	The password to use to log into the CMS.
Authentication mode	The authentication type used by the CMS.

3. Click **Connect**.

The software attempts to connect to the CMS using the specified information. If the authentication is successful, the Workbench window is displayed.

2.2 Workbench window

By default, the Workbench user interface consists of several primary elements:

- Menu bar
- Toolbar
- Project explorer

The Workbench project explorer contains the current project, and all jobs and datastores that are a part of it.

- **Workspace**

The workspace is the area of the Workbench window where you define, display, and modify objects. When you create a new object, the Workbench automatically opens a new editor tab in the workspace. The workspace is also used to display the data of a source table and to display the job status and execution logs.

Workspace editors

The workspace area of the Workbench contains tabs for the editors you currently have open. The editors that may appear in the workspace include the following:

- **Replication job editor**

The replication job editor allows you to display and edit replication job options. For example, the target table schema, projections expressions and filters, and replication groups. The name of the open replication job is indicated in the tab's title. If the replication job contains unsaved data, it is indicated with an asterisk (*).

- **Datastore editor**

The datastore editor allows you to display and edit datastore properties, browse and import metadata, and display and edit table properties. The name of the open datastore is indicated in the tab's title. If the datastore contains unsaved data, it is indicated with an asterisk (*).

- **Monitoring editor**

The monitoring editor displays your replication jobs' status, execution progress, and any applicable logs. Trace, monitor, and error logs are available.

Related Topics

- [Replication job editor](#)
- [Datastore editor](#)
- [Monitoring editor](#)

2.3 Menu bar

This section contains a brief description of the Workbench menus.

2.3.1 File menu

The File menu contains standard Windows as well as software-specific options.

Option	Description
New	Define a new folder, project, datastore, or replication job.
Open File	Open an existing Workbench file.
Close	Close the active Workbench editor.
Close All	Close all open editors.
Save	Save the object in the active editor.
Save All	Save all objects in open editors.
Exit	Exit the Workbench.

2.3.2 Edit menu

The Edit menu contains standard Windows options.

Option	Description
Undo	Undo the last operation.
Redo	Redo the last undone operation.
Cut	Cut the selected objects or text and place it on the clipboard.
Copy	Copy the selected objects or text to the clipboard.
Paste	Paste the contents of the clipboard into the active editor or text box.
Delete	Delete the selected objects or text.
Select All	Select all objects in the active editor or all text in the active text box.

2.3.3 Tools menu

The Tools menu contains replication job editor and datastore editor options.

Option	Description
Open monitoring editor	Open the job monitoring editor.
Export replication job	Export a selected replication job to the local filesystem. Note: This option is available only for the replication job editor.
Deploy	Deploy a replication job or datastore definition to a Data Services repository.
Execute	Deploy a replication job to a Data Services repository and execute the job. Note: This option is available only for the replication job editor.

2.3.4 Validate menu

The Validate menu contains replication job editor and datastore editor validation options.

Option	Description
Validate	Validate the active Workbench replication job for errors. Note: The Validate option is available only for the replication job editor.
Show DDL	Display the data definition language (DDL) for the active Workbench object. Note: The Show DDL option is available only for the replication job editor.
Show ATL	Display the ATL for the active Workbench object. Note: Do not use this option to create a file for import into Data Services. Use the Tools Export Replication Job option to export a replication job.

2.3.5 Window menu

The Window menu provides software-specific window options.

Option	Description
Error Log	Display the Error Log in the workspace area.
Problems	Display the Problems tab in the workspace area.
Project Explorer	Display the project explorer.
Reference Analysis	Display a reference impact analysis for the active Workbench object in the workspace area.
Other	Display other window panes.
Reset Perspective	Reset the Workbench interface perspective to its default settings.
Preferences	Display the Preferences window.

2.3.6 Help menu












The Help menu provides standard help options.

Option	Description
Welcome	Display the Workbench welcome screen.
Help	Display the SAP BusinessObjects Data Services Workbench Guide.
About	Display information about the software including the version of the Workbench and copyright information.
Key Assist	Display information about keyboard shortcuts available in the Workbench.

2.4 Toolbar

In addition to many of the standard Windows tools, the software provides application specific tools, including:

Icon	Tool	Description
	New	Define a new folder, project, datastore, or replication job.

Icon	Tool	Description
	Save	Save the object in the active editor.
	Cut	Cut the selected objects or text and place it on the clipboard.
	Copy	Copy the selected objects or text to the clipboard.
	Paste	Paste the contents of the clipboard into the active editor or text box.
	Delete	Delete the selected objects or text.
	Open monitoring editor	Open the job monitoring editor.
	Validate	Validate the active Workbench resource for errors.
	Show DDL	Display the data definition language (DDL) for the active Workbench resource.
	Deploy ATL	Deploy the ATL for the active replication job to the Data Services repository.
	Execute	Deploy the ATL for the active replication job to the Data Services repository, and execute the replication job.
	Quick replication wizard	Start the quick replication wizard to easily define source and target datastores and create a basic replication job.

2.5 Project explorer

The project explorer contains the projects currently defined in the user workspace. Each Workbench project is represented as a folder containing all datastores and replication jobs associated with that project.

When you create new projects, datastores, and replication jobs, they are displayed in the project explorer. You can also use the project explorer to import and export Workbench project information.

2.6 Replication job editor

The Workbench replication job editor opens in the workspace when you create a new replication job or open an existing replication job from the project explorer.

The replication job editor consists of several areas:

- Source datastore pane

Displays the name of the source datastore and all imported tables.

- Target datastore pane

Displays the name of the target datastore and all mapped tables and columns. In this pane, mapped objects may be grouped into replication groups.

- Replication job configuration pane

When you select an object such as a datastore table or column in the replication job editor, its configuration options are displayed in this pane.

The option groups that are available depend on the type of object that is selected:

Object	Available options
Source table	Properties
Replication job	Properties Variables Scripts
Replication group	Properties
Target table	Properties Columns Keys Indexes Options Filter data DDL Options Delta load
Column	Properties

You can adjust the size of each pane by dragging the borders between them. In addition, you can hide or reveal the replication job configuration pane by clicking the arrows in the center of its border.

Related Topics

- [Using the replication job editor](#)

2.7 Datastore editor

The Workbench datastore editor opens in the workspace when you create a new datastore or open an existing datastore from the project explorer or replication job editor.

The datastore editor consists of these areas:

- Navigation pane

The navigation pane displays the configuration and imported tables and views of the datastore. For SAP datastores, the navigation pane also displays the imported SAP extractors.

When the datastore configuration is active, you can edit the datastore properties and settings. When the datastore tables, extractors, or views are active, you can import or re-import tables, extractors, and views, view data, and compare imported metadata with the metadata in the database.

- Object properties pane

Displays tabs for the properties and options for the object currently selected in the navigation pane. The option tabs that are available depend on the type of object that is selected:

Object	Available option
Table	Properties Columns Keys Indexes Partitions Attributes
Column	Properties Attributes

You can adjust the size of each pane by dragging the borders between them. In addition, you can hide or reveal the navigation pane by clicking the arrow in the center of its border.

Related Topics

- [Using the datastore editor](#)

2.8 Monitoring editor

The Workbench monitoring editor opens in the workspace when you click the icon in the toolbar or select it from the **Tools** menu.

The monitoring editor consists of several areas:

- Job selection toolbar

Displays the Data Services jobs available in the repository to which the Workbench is connected and allows you to execute them.

Note:

The monitoring editor displays all standard Data Services jobs in the repository, not Workbench-specific replication jobs.

- Execution history pane

Displays the execution history for the selected job. Each line indicates the current status, execution date, and duration of a single job execution instance.

- Execution history dashboard pane

Displays a graphical representation of the history for the selected job. You can change the type of chart by choosing from the drop-down box in the upper right corner of the pane.

- Execution details pane

Displays the details for the individual job execution instance selected in the execution history pane. You can view the trace, monitor, and error logs for the execution instance, as well as the execution statistics.

You can adjust the size of each pane by dragging the borders between them. In addition, you can hide or reveal the dashboard and details panes by clicking the arrows in the center of their borders.

Database Migration

Migrating data and schema information between different database systems is a complex task that can take days or even months depending on the type of database systems, the complexity of the database schema, and the size of the data set being migrated. In addition, incompatibilities between the source and target database types could require manual schema and data corrections.

The Data Services Workbench simplifies this database migration process considerably. In the Workbench, you provide connection information for the source and target databases and select the tables that you want to migrate. The Workbench automatically creates Data Services jobs, work flows, and data flows and imports them into a Data Services repository.

After you have created the replication jobs, you can use the Workbench to execute the jobs and monitor their progress. If required, you can also edit the generated data flows in the Designer, and use other Data Services tools to schedule and execute the jobs.

3.1 Migrating data and schema information

The process of migrating data and schema information from one database to another with the Workbench generally consists of a few basic steps:

1. Create a project in the project explorer.
2. Create a source datastore definition.
3. Create a target datastore definition.
4. Create a replication job using your source and target datastores.
 - a. Select the source objects that you want to migrate.
 - b. Group the target objects and specify the order of migration.
 - c. Adjust the target schema as required.
 - d. Specify the replication behavior for the job.
5. Deploy and execute the replication job.

You can choose to perform the datastore and replication job definition steps manually with the standard individual datastore and replication job editors, or you can complete the process using the quick replication wizard. If you use the replication wizard, you can modify your replication job by using the standard editors after the wizard completes.

Related Topics

- [Project explorer](#)

- [Datastore editor](#)
- [Replication job editor](#)

3.1.1 Creating a Workbench project

If you want to create a new project, start the new project wizard in one of the following ways:

- Right-click inside the project explorer and choose **New > Project**.
- Choose **New > Project** from the **File** menu.
- Click the **New** toolbar icon. Choose **General > Project** and click **Next**.

1. Specify the name for the new project and click **Next**.

Note:

The project name can contain only alphanumeric characters and the underscore character.

2. If you want to save the files for the project to a different location, uncheck **Use default location** and specify a location for the project files.
3. Click **Finish** to create the project.

The new project is displayed in the project explorer.

You can now add Workbench objects such as datastores and replication jobs to your project.

3.1.2 Using the quick replication wizard

You can use the quick replication wizard to easily migrate source data to an SAP HANA, Sybase IQ, or Teradata target database.

1. Click the **Quick replication wizard** icon in the toolbar.

The quick replication wizard is displayed.

2. Specify the name for the new replication project and click **Next**.

Note:

The project name can contain only alphanumeric characters and the underscore character.

The source datastore configuration screen is displayed.

3. Specify the settings for the source datastore and click **Next**.

- a. Select the datastore type.

The available configuration settings for the selected database or application type are displayed.

- b. Set the datastore configuration settings as required for your database or application.

Note:

Required settings are marked with an asterisk (*), and required settings that are missing a value are highlighted in red in the **Property name** column. Settings that have changed from the default value are listed in bold.

- c. Click **Test connection** to verify that the connection settings are valid.
If the connection test fails, correct the datastore connection settings and re-test the connection.

Note:

The connection test is between the database or application and the Central Management Console (CMC) host, not the Workbench host. Ensure that the connection settings between the CMC and datastore database or application are valid.

The source object selection screen is displayed.

4. Select the objects that you want to replicate to the target datastore and click **Next**.

The target datastore configuration screen is displayed.

5. Specify the settings for the target datastore and click **Next**.

- a. Select the datastore type.

The available configuration settings for the selected database or application type are displayed.

- b. Set the datastore configuration settings as required for your database or application.

Note:

Required settings are marked with an asterisk (*), and required settings that are missing a value are highlighted in red in the **Property name** column. Settings that have changed from the default value are listed in bold.

- c. Click **Test connection** to verify that the connection settings are valid.
If the connection test fails, correct the datastore connection settings and re-test the connection.

The repository and job server selection screen is displayed.

6. If you want to execute the replication job without making any further changes, select **Execute replication job now**.

- a. In the left column, select the Data Services repository to use.

The jobs generated by the Workbench will be stored in this repository. By default, the first repository in the list is selected.

- b. In the right column, select the Data Services Job Server to use to execute the replication job.
By default, the Job Server associated with the selected repository is used.

7. Click **Finish** to save the replication job and execute it with the selected settings, if applicable.

If you chose to execute the replication job, the monitoring editor is displayed.

The Workbench saves the replication job and datastore definitions to the local filesystem. If you chose to execute the replication job now, it also creates Data Services jobs and datastores for the replication job, imports them to the selected repository, and executes them using the selected Job Server.

3.1.3 Using the datastore editor

The datastore editor provides a powerful interface that allows you to fine-tune your datastore definition, including importing database tables, inserting or modifying columns, keys, indexes, and partitions, modifying attributes, and so on.

Tip:

The datastores defined and used in the Workbench are standard Data Services datastores. For more information about Data Services datastores, see the *Reference Guide*.

If you have already created a datastore, open it in the datastore editor by double-clicking the datastore name in the project explorer.

If you want to create a new datastore definition, start the new datastore wizard in one of the following ways:

- Right-click your project in the project explorer and choose **New > Datastore**.
- Select your project in the project explorer and choose **New > Datastore** from the **File** menu.
- Select your project in the project explorer and click the **New** toolbar icon. Choose **Data Services > Datastore** and click **Next**.

1. Specify the name for the new datastore and click **Next**.

Note:

The datastore name can contain only alphanumeric characters and the underscore character.

2. Specify the settings for the datastore and click **Next**.

- a. Select the datastore type.

The available configuration settings for the selected database or application type are displayed.

- b. Set the datastore configuration settings as required for your database or application.

Note:

Required settings are marked with an asterisk (*), and required settings that are missing a value are highlighted in red in the **Property name** column. Settings that have changed from the default value are listed in bold.

- c. Click **Test connection** to verify that the connection settings are valid.

If the connection test fails, correct the datastore connection settings and re-test the connection.

The object selection screen is displayed.

3. Select the objects that you want to import to the datastore definition and click **Finish**.

The new datastore is opened in the datastore editor.

You can now use the datastore editor to further customize your datastore definition. For example, you can select a table or view and modify its properties, columns, keys, and so on, or use the configurations section to add additional configurations to your datastore.

Related Topics

- [Datastore editor](#)
- [Supported data migration sources and targets](#)
- [Reference Guide: Objects, Descriptions of objects, Datastore](#)

3.1.4 Using the replication job editor

The replication job editor provides a powerful interface that allows you to fine-tune your replication job, including filtering datastore tables, customizing target replication groups, modifying table and column mappings, and so on.

Note:

The replication jobs defined and used in the Workbench are defined in Workbench-specific metadata. When you deploy the replication job to a Data Services repository, the Workbench uses this metadata to create standard Data Services objects such as job, work flows, and data flows. For more information about Data Services objects, see the *Reference Guide*.

If you have already created a replication job, open it in the replication job editor by double-clicking the replication job name in the project explorer.

If you want to create a new replication job, start the new replication job wizard in one of the following ways:

- Right-click your project in the project explorer and choose **New > Replication Job**.
 - Select your project in the project explorer and choose **New > Replication Job** from the **File** menu.
 - Select your project in the project explorer and click the **New** toolbar icon. Choose **Data Services > Replication job** and click **Next**.
1. Specify the name for the new replication job and click **Next**.

Note:

The replication job name can contain only alphanumeric characters and the underscore character.

2. Click ... to specify the source and target datastores and click **Finish**.

A list of available datastores is displayed. If you have not already created the datastores, you can finish the new replication job wizard and define the datastores later.

The new replication job is opened in the replication job editor.

You can now use the replication job editor to further customize your replication job definition. For example, you can change the replication behavior, modify the target schema, customize column mappings, and so on.

Related Topics

- [Replication job editor](#)
- [Reference Guide: Objects](#)

3.1.4.1 Modifying replication job properties

When you open a replication job or select the root replication job node in the target datastore pane, the configuration area displays several tabs that you can use to configure the way that the Workbench processes the replication job:

Tab	Description
Properties	Displays the name, owner, replication behavior and description for the replication job.
Variables	Displays information about the variables available for use in the replication job. You can add or delete variables, as well as alter variable properties such as data type, length, precision, scale, and default value.
Scripts	Displays the pre-load and post-load scripts for the replication job. You can modify the scripts directly in the text box or by clicking Edit to launch the Workbench expression editor. You can also validate the scripts.

Related Topics

- [Using the expression editor](#)

3.1.4.1.1 Replication behavior

When you deploy and execute a Workbench replication job, the Workbench automatically generates the ETL metadata for the job and deploys it to a Data Services repository. The metadata that is generated depends on the replication behavior setting for the replication job.

The following replication behavior options are available:

- Schema and data

In the schema and data scenario, the Workbench assumes that the tables do not exist in the target database and need to be created. As a result, the ETL job metadata that is generated includes the following objects:

- Source and target datastores
 - Source and target tables
 - A Data Services job that contains both the script to create the target tables, as well as the work flows and data flows that move data from the source tables to the target tables.
- Data only

In the data-only scenario, the Workbench assumes that the tables already exist in the target database. As a result, the ETL job metadata that is generated includes the following objects:

- Source and target datastores
- Source and target tables
- A Data Services job that contains work flows and data flows that move data from the source tables to the target tables.

Note:

Schema and data is the default replication behavior.

3.1.4.2 Adjusting the target schema

3.1.4.2.1 Modifying target table properties

When you select a table in the target datastore pane, the configuration area displays several tabs that you can use to configure the way that the Workbench processes that table:

Tab	Description
Properties	Displays the table name, owner, and description and provides the ability to view the data in the source table.
Columns	Displays information about the target table columns. After the table has been imported, you can insert and delete columns from the target table. You can also change column names, data types, descriptions, and so on, as well as alter the column mappings with the expression editor.
Keys	Displays information about the table's primary and foreign keys, if any. After the keys have been imported, you can add, delete, and modify them.
Indexes	Displays information about the table's indexes. The primary index is listed first and followed by any secondary indexes. You can add or remove indexes, as well as add or remove columns from existing indexes.

Tab	Description
Options	<p>Displays options that Data Services uses to read the source table and load the target table.</p> <p>For more information about the available source and target options, see the <i>Reference Guide</i>.</p> <p>Note: Only options used by the Workbench are displayed.</p>
Filter data	<p>Allows you to filter the rows of the source table.</p> <p>You can use basic filter editor or the expression editor to modify the filter.</p>
DDL Options	<p>Allows you to specify the behavior of DDL generation.</p> <p>For example, whether to create foreign keys or indexes.</p>
Delta load	<p>Displays options that modify the delta load behavior for a table.</p> <p>For more information about the available delta load options, see Delta load migration.</p>

Related Topics

- [Reference Guide: Objects, Descriptions of objects, Source](#)
- [Reference Guide: Objects, Descriptions of objects, Target](#)

3.1.4.2.2 Grouping target tables

Many data warehouse tables have referential integrity constraints such as primary key and foreign key relationships. The primary key table needs to be loaded before the foreign key table if there is a relationship between two tables.

You can avoid referential integrity issues by using the replication job editor to group tables. For example, you can put the primary key tables in the first group and foreign key tables in the second group. In another scenario, you could put dimension tables in the first group and fact tables in the second group.

By default, tables are added to the Default_Group that is created in a new replication job. However, there is no limit to the number of replication groups supported in a Workbench replication job.

To group tables in the replication job editor:

1. Create a new replication group in the target schema.
 - a. Right-click on the replication job root node in the target schema pane and choose **New replication group**.
A new replication group is added to the target schema.
 - b. Assign a meaningful name to the replication group by highlighting the new replication group and clicking it again.

2. Assign tables to the new group by dragging them from the source pane to the replication group.

Note:

A table may belong to only one group in the target schema. You can create multiple target tables from a single source table, but each target table must be named uniquely.

3. Use the up and down arrows in the target schema to order the replication groups as needed for your requirements. When the replication job is executed, the groups will be migrated starting from the top of the list.

You can also rearrange replication groups by dragging them within the target schema.

Tip:

To automatically order the tables in the target schema by their dependencies, click **Auto order**.

Note:

Replication groups can be sequential or parallel. In a sequential group, the dataflows generated by the Workbench will be executed in sequential order. In a parallel group, the dataflows are executed in parallel. To change the group type, right-click on the group and choose **Parallel** or **Sequential** from the **Change replication group type** menu.

3.1.4.2.3 Changing column data types

Data Services supports conversion to and from its internal data types and performs data type conversions when it imports metadata from external sources or targets into the repository, as well as when it loads data into an external table. When the Workbench creates target tables, the software converts from internal data types to the data types used by the target database.

For more information about internal data type conversions in Data Services, see the *Reference Guide*.

The Workbench allows you to overwrite the default data conversion rules. To change the data type for a column:

1. Expand the table containing the column in the target schema pane.
2. Select the target column.

The column details are displayed in the configuration pane.

3. Select the new data type for the column from the **Data type** drop-down box.

If required for the new data type, set the **Length**, **Precision**, and **Scale** for the column.

Related Topics

- [Reference Guide: Data Types, Data type conversion, Conversion to or from internal data types](#)

3.1.4.3 Filtering datastore tables

In a replication job, you might want to migrate only a subset of the rows from a particular source table to your target datastore.

To migrate a subset of table rows, use a filter on the datastore table in the replication job editor:

1. Select the table that you want to filter in the target schema pane.
2. Click the **Filter data** tab in the configuration pane.
3. For basic filtering, use the table filters configuration window.

- a. Select **Basic** and click **Edit filter**.

The table filters configuration window is displayed.

- b. Drag the columns you want to use to filter from the columns pane to the filters pane.

The columns are displayed in the filters panel and grouped together with a logical **AND** or **OR** grouping operator.

- c. For each column, define the filter logic by specifying the comparison operator and value.

Tip:

You can create more complex, nested filter expressions by dragging the columns on top of each other. The nesting layout is displayed in a graphical preview.

- d. Change the grouping logic as required by double clicking the grouping operator.
- e. To remove a filter, click the column or grouping operator to highlight the filter and click the **Remove selected filter** button.
- f. When you are satisfied with the filter, click **OK**.

The filter is displayed in the **Filter data** tab.

4. For more advanced filtering, use the Workbench expression editor to define the filter.

- a. Select **Advanced** and click **Edit filter**.

The expression editor is displayed.

- b. Complete your filtering expression and click **OK**.

The filter expression is displayed in the **Filter data** tab.

- c. To validate your filter expression, click **Validate**.

Related Topics

- [Using the expression editor](#)

3.1.5 Deploying and executing the replication job

When you are satisfied with your replication job configuration, you can deploy it to a Data Services repository and execute it.

1. Double-click the replication job in the project explorer to open it in the replication job editor.
2. Click **Validate** to verify that the replication job is free of errors.
3. If you want to deploy the job but not execute it, click **Deploy**.

- a. Select the repository where you want to deploy the replication job and click **OK**.

A progress bar indicating the deployment progress is displayed.

A message indicating the success or failure of the deployment is displayed. If any errors occur during deployment, the message includes any details available.

4. If you want to deploy and execute the job immediately, click **Execute**.

- a. Select the repository where you want to deploy the replication job and click **Next**.

The job execution parameters screen is displayed.

- b. Specify whether to execute the job as an initial load or delta load.
- c. Specify the job server and other execution options to use when executing the job in the **Execution options** tab.
- d. If you want to include additional information in the trace log, change the values in the **Trace** tab to **Yes** as required.
- e. Click **Finish** to deploy and execute the job.

A progress bar indicating the deployment progress is displayed. If you want to halt the deployment, click the **Cancel operation** button next to the progress bar.

The monitoring editor for the job is displayed.

Related Topics

- [Monitoring editor](#)

3.1.5.1 Error recovery

If an error occurs while your replication job is being executed, only some rows may be inserted into a target table.

There are several methods that you can use to prevent duplicate rows from being inserted into the target table, including:

- Data flow design

You can design the replication job to completely replace the target table each time the job is executed. You might choose to use this method if the number of rows changed in the target table is a significant percentage of the target table. You can also use techniques such as bulk loading options to improve the overall performance of the replication job.

- Auto-correct load target table option

You can use the auto-correct load option to check the target table for existing rows before new rows are added to the table. However, this option can needlessly slow jobs that are not executed in recovery mode. You might choose to use this method if the number of rows changed in the target table is a relatively small percentage of the target table.

- Pre-load SQL commands

You can use pre-load SQL commands to remove partial database updates that have occurred during an incomplete job execution. Typically, the pre-load SQL command deletes rows based on a variable set before the partial insertion began.

3.2 Delta load migration

Delta load jobs move rows that have been added or modified since the last time the job was executed. There are many reasons to implement a delta load job, but the most common is to reduce the time the loading process takes. Instead of loading millions of rows each time the job is run, you can process only the few that have changed. Another reason could be to maintain historical data; you might want to keep the old data in your data warehouse and add the current state so that you can see the changes over time.

3.2.1 Delta load jobs in the Workbench

When you create a replication job that has a table configured with a delta load method, the Workbench automatically creates two sets of metadata: a first load job and a delta load job. The first load job performs a full replication, with any user-defined filters applied, from the source datastore to the target datastore. The delta load job can be used to capture only changed data after the first load.

The Workbench creates two scripts within the delta load job: the pre-load script and the post-load script. These scripts keep track of the time each job is executed so that the next execution can continue from where the last one ended. You can customize the pre-load and post-load scripts in the **Scripts** property tab for the replication job.

To help keep track of the delta job executions, the Workbench maintains an internal table in the target datastore that is created after the successful execution of the first load job. Because of this, the full replication job must be successfully executed at least once before the delta load job can be executed.

Note:

By default, all target tables within the Workbench delta load job are created with the **Auto correct load** option enabled. For more information about this option, see the “Target tables” section in the *Reference Guide*.

Delta job execution

During execution, the delta job first determines the last time that it was executed, if ever. It uses the last execution time in a filter within the data flows to select only those records that are new. After the data flows have completed successfully, the delta job updates the internal metadata with the current time, so that any future delta execution starts from that time.

Related Topics

- [Reference Guide: Objects, Descriptions of objects, Target tables](#)

3.2.2 Delta load options for tables

For each table, the Workbench supports several delta load options:

- No delta load

The Workbench skips this table in the delta load.

- Reload the full table

The Workbench performs a complete refresh of the table.

- Use timestamp column

The Workbench uses a timestamp column or columns from the table to perform the delta load. Data Services uses the timestamp from each load to extract only new data from after the last execution.

If a table has more than one datetime or timestamp column, you must choose the column to use for the delta load process. If a table does not contain a datetime or timestamp column, this option is disabled. If you use a column that contains only the date, you can also choose to use an additional column that specifies the time.

- Use CDC

The Workbench uses the changed data capture (CDC) functionality provided by a supported SAP application to extract only the data that has changed.

If the datastore does not support CDC functionality, this option is disabled.

- Use custom filter

The Workbench uses a user-defined filter to determine the start time used to perform the delta load.

Tip:

In the Workbench, the delta load job works best with tables that use a timestamp column as the primary key.

To change the delta load method for a table, right-click the table in the target schema pane and choose the delta load method under **Change delta load support**. You can also select the table in the target schema pane and choose the delta load method in the **Delta load** tab of the replication job editor.

3.2.3 Delta load options for SAP applications

For applications that support changed data capture (CDC) functionality, the Workbench automatically extracts changed data for the delta load.

You can use the Workbench-generated variable `$END_TIME` to specify the time frame for which to extract changed data:

- If the value of `$END_TIME` is not specified, the Workbench uses the current system time for its value. Each delta load will extract new data from the last execution to the current date-time.
- If you want to continue from a previous execution, retrieve the timestamp of the execution you want to recover from your application and assign it to `$END_TIME` in the replication job pre-load script.

Note:

The Workbench treats applications that do not support CDC as tables for the delta load.

Additional Information

4.1 Supported data migration sources and targets

Supported sources

The Workbench supports the following database and application types as migration sources:

- Data Services-supported databases
 - Attunity Connector
 - HP Neoview
 - DB2
 - SAP HANA
 - Informix
 - Microsoft SQL Server
 - MySQL
 - Netezza
 - ODBC
 - Oracle
 - Sybae ASE
 - Sybase IQ
 - Teradata
- SAP Applications
- SAP BW Source

Note:

For more information about the available options for each supported datastore source type, see the *Reference Guide*.

Supported targets

The Workbench supports the following database types as migration targets:

- SAP HANA
- DB2
- Oracle
- Sybase IQ
- Teradata
- Sybase ASE
- Microsoft SQL Server

Note:

For more information about the available options for each supported datastore target type, see the *Reference Guide*.

Related Topics

- [Reference Guide: Objects, Descriptions of objects, Datastore](#)

4.2 Using the expression editor

The expression editor allows you to use variables, functions, and basic logic structures in many areas of a replication job. For example, you can use the expression editor to customize the mapping of a target column or the replication job's pre-load or post-load scripts.

4.2.1 Supported operators

This section describes the operators that can be used in Workbench expressions and filters.

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
=	Comparison, equal to
<	Comparison, less than
<=	Comparison, less than or equal to
>	Comparison, greater than
>=	Comparison, greater than or equal to
!=	Comparison, not equal to
	Concatenate
%	Modulus Returns the remainder when one number is divided by another.
AND	Logical AND

Operator	Description
OR	Logical OR
NOT	Logical NOT
IS NULL	Comparison, is a NULL value
IS NOT NULL	Comparison, is not a NULL value

4.2.2 Including functions in expressions

In Data Services, functions take input values and produce a return value. Input values can be parameters passed into a data flow, values from a column of data, or variables defined inside a script. This section discusses functions and how to include them in a Workbench replication job.

4.2.2.1 Built-in functions

This section describes each built-in function available in Data Services.

The following tables list the names and descriptions of functions, grouped by each function's category in the Workbench expression editor.

For a complete listing of all built-in functions and their syntax, parameters, and return values, see “Functions and Procedures, Descriptions of built-in functions” in the *Reference Guide*.

Aggregate functions

Function	Description
avg	Calculates the average of a given set of values.
count	Counts the number of values in a table column.
count_distinct	Count the number of distinct non-null values in a table column.
max	Returns the maximum value from a list.
min	Returns the minimum value from a list.
sum	Calculates the sum of a given set of values.

Conversion functions

Function	Description
cast	Returns a value in the cast data type.
extract_from_xml	Extracts XML directly from single column in a database table, and converts it into its internal nested relational data model (NRDM).
interval_to_char	Returns a string representation of the interval.
julian_to_date	Converts a Julian value to a date.
load_to_xml	Generates XML text from NRDM and loads it into a single database column (Assumes the database supports XML text in its columns).
long_to_varchar	Converts a data type from long to varchar.
num_to_interval	Converts a numeric value to an interval.
to_char	Converts a date or numeric type to a string.

Function	Description
to_date	Converts a string to a date.
to_decimal	Converts a varchar to a decimal.
varchar_to_long	Converts a data type from varchar to long.

Cryptographic functions

Function	Description
decrypt_aes	Decrypts the input string using the user-specified passphrase and key length using the AES algorithm.
encrypt_aes	Encrypts the input string using the user-specified passphrase and key length using the AES algorithm.

Database functions

Function	Description
key_generation	Generates keys for the specified table, after determining the appropriate starting value.
sql	Runs a SQL operation in the specified database.
total_rows	Returns the number of rows in a particular table in a datastore.

Date functions

Function	Description
add_months	Adds a given number of months to a date.
concat_date_time	Returns a datetime from separate date and time inputs.
date_diff	Returns the difference between two dates or times.
date_part	Extracts a component of a given date.
day_in_month	Determines the day in the month on which the given date falls.
day_in_week	Determines the day in the week on which the given date falls.
day_in_year	Determines the day in the year on which the given date falls.
fiscal_day	Converts a given date into an integer value representing a day in a fiscal year.
isweekend	Indicates that a date corresponds to Saturday or Sunday.
julian	Converts a date to its integer Julian value, the number of days between the start of the Julian calendar and the date.
last_date	Returns the last date of the month for a given date.
month	Determines the month in which the given date falls.

Function	Description
quarter	Determines the quarter in which the given date falls.
sysdate	Returns the current date as listed by the Job Server's operating system.
systime	Returns the current time as listed by the Job Server's operating system.
week_in_month	Determines the week in the month in which the given date falls.
week_in_year	Determines the week in the year in which the given date falls.
year	Determines the year in which the given date falls.

Environment functions

Function	Description
get_env	Returns a value for the specified environmental variable.
get_error_filename	Returns the full path and file name for the error log.
get_monitor_filename	Returns the full path and file name for the monitor log.
get_trace_filename	Returns the full path and file name for the trace log.
is_set_env	Verifies if the specified environment variable is set.
set_env	Sets an environmental variable temporarily to a specified value.

Lookup functions

Function	Description
lookup	Finds a value in one table or file based on values in a second table or file.
lookup_ext	Finds data from a database table, flat file, or memory datastore table.

Math functions

Function	Description
abs	Returns the absolute value of an input number.
ceil	Returns the smallest integer value greater than or equal to an input number.
floor	Returns the largest integer value less than or equal to an input number.
ln	Returns the natural logarithm of the given numeric expression.
log	Returns the base-10 logarithm of the given numeric expression.
mod	Returns the remainder when one number is divided by another.
power	Returns the value of the give expression to the specified power.
rand	Returns a random number between 0 and 1.
rand_ext	Returns a random number between 0 and 1.

Function	Description
round	Rounds a given number to the specified precision.
sqrt	Returns the square root of the given expression.
trunc	Truncates a given number to the specified precision.

Miscellaneous functions

Function	Description
base64_decode	Returns the source data after decoding the base64-encoded input.
base64_encode	Returns the base64-encoded data in the engine locale character set.
current_configuration	Returns the name of the datastore configuration in use at runtime.
current_system_configuration	Returns the name of the system configuration in use at runtime. If no system configuration is defined, returns a NULL value.
dataflow_name	Returns the data flow name in which this call exists. If the call is not in a data flow, returns NULL.
datastore_field_value	Retrieves the value of a specified datastore field.
db_database_name	Returns the database name of the datastore configuration in use at runtime.
db_owner	Returns the real owner name for the datastore configuration that is in use at runtime.
db_type	Returns the database type of the datastore configuration in use at runtime.
db_version	Returns the database version of the datastore configuration in use at runtime.
decode	Returns an expression based on the first condition in the specified list that evaluates to TRUE.
file_exists	Checks to see if a given file or directory exists.
gen_row_num	Returns an integer value beginning with 1 then incremented sequentially by 1 for each additional call. This function can be used to generate a column of row IDs.
gen_row_num_by_group	Returns group row number of the record.
gen_uuid	Returns a unique varchar string identifier.
get_domain_description	Returns the description of a value when given the domain name and the value.
get_file_attribute	Returns date created, date modified, or size (in bytes) of a physical file.
greatest	Returns greatest of the list of one or more expressions.
host_name	Returns the name of the computer on which the job is executing.
ifthenelse	Allows conditional logic in mapping and selection operations.

Function	Description
is_group_changed	Returns 1 if the group is changed, 0 otherwise.
isempty	Indicates if a nested table contains data.
job_name	Returns the name of the job in which the call to this function exists.
job_run_id	Retrieves the job run ID for the current job execution.
least	Returns the least in a list of one or more expressions.
nvl	Replaces NULL values.
previous_row_value	Returns the column value of previous row.
pushdown_sql	Allows you to create dynamic WHERE clauses.
raise_exception	Calling this function causes an exception to be generated.
raise_exception_ext	Same as raise_exception, but takes a second parameter for an exit code.
repository_name	Returns a database connection string and owner name. For example: beq-local.DBUser. This is the ID for the repository from which the job is run.
sleep	Suspends the execution of the data flow or work flow from where it is called.
system_user_name	Returns the user name used to log into the Job Server's operating system.
table_attribute	Retrieves the value of a specified table attribute.

Function	Description
truncate_table	Allows you to explicitly expunge data from a memory table.
wait_for_file	Returns the existing files that match the input file pattern.
workflow_name	Returns the work flow in which this call exists. Returns the name of the inner most work flow in cases where several work flows enclose this function call. If no work flow is found, returns job name.

SAP functions

Function	Description
sap_openhub_process-chain_execute	Starts the process chain that extracts data from an SAP NetWeaver Business Warehouse(BW) and loads the extracted data into an Open Hub Destination table.
sap_openhub_set_read_status	Sends the read status for the Open Hub table to SAP NetWeaver BW.

String functions

Function	Description
ascii	Returns the decimal value of the first character for the given string using ASCII character set. If the character passed is not a valid ASCII character, -1 is returned.
chr	Get character representation of provided ASCII value.
double_metaphone	Encodes the input string using the Double Metaphone algorithm and returns a string.
index	Returns the index of a given word in a string.
init_cap	Changes the characters in a string to title case.
length	Returns the number of characters in a given string.
literal	Returns an input constant expression without interpolation. Allows you to assign a pattern to a variable without interpolation.
lower	Changes the characters in a string to lowercase.
lpad	Pads a string with characters from a specified pattern.
lpad_ext	Pads a string with logical characters from a specified pattern.
ltrim	Removes specified characters from the start of a string.
ltrim_blanks	Removes blank characters from the start of a string.
ltrim_blanks_ext	Removes blank and control characters from the start of a string.
match_pattern	Matches whole input strings to simple patterns supported by Data Services. This function does not match substrings.

Function	Description
match_regex	Matches whole input strings to the pattern that you specify with regular expressions (regular expressions based on the POSIX standard) and flags. This function does not match substrings.
match_simple	
print	Prints the given string to the trace log.
replace_substr	Returns a string where every occurrence of a given search string in the input is substituted by the given replacement string.
replace_substr_ext	Takes an input string, replaces specified occurrences of a specified sub-string with a specified replacement and returns the result. You can also use this function to search for hexadecimal or reference characters.
rpad	Pads a string with characters from a given pattern.
rpad_ext	Pads a string with logical characters from a given pattern.
rtrim	Removes given characters from the end of a string.
rtrim_blanks	Removes blank characters from the end of a string.
rtrim_blanks_ext	Removes blank and control characters from the end of a string.
search_replace	Searches input parameters and replaces by matching criteria and values specified by search table.
soundex	Encodes the input string using the Soundex algorithm and returns a string. Use when you want to pushdown the function to the database-level.
substr	Returns a specific portion of a string starting at a given point in the string.
upper	Changes the characters in a string to uppercase.
word	Returns one word out of a string.
word_ext	Returns the word identified by its position in a delimited string.

System functions

Function	Description
exec	Sends a command to the operating system for execution.
mail_to	Sends the specified e-mail message.
smtp_to	Sends the specified e-mail message using the SMTP protocol.

User functions

You can create your own functions by writing script functions in SAP BusinessObjects Data Services scripting language using the Designer smart editor.

For more information about user-defined custom functions, see “Functions and Procedures, Custom functions” in the *Reference Guide*.

Validation functions

Function	Description
is_valid_date	Indicates if an expression can be converted into a valid date value.
is_valid_datetime	Indicates if an expression can be converted into a valid datetime value.
is_valid_decimal	Indicates if an expression can be converted into a valid decimal value.
is_valid_double	Indicates if an expression can be converted into a valid double value.
is_valid_int	Indicates if an expression can be converted into a valid integer value.
is_valid_real	Indicates if an expression can be converted into a valid real value.
is_valid_time	Indicates if an expression can be converted into a valid time value.

Related Topics

- [Reference Guide: Functions and Procedures, Descriptions of built-in functions](#)
- [Reference Guide: Functions and Procedures, Custom functions](#)

4.3 Managing Workbench project information

4.3.1 Creating a Workbench project

If you want to create a new project, start the new project wizard in one of the following ways:

- Right-click inside the project explorer and choose **New > Project**.
- Choose **New > Project** from the **File** menu.
- Click the **New** toolbar icon. Choose **General > Project** and click **Next**.

1. Specify the name for the new project and click **Next**.

Note:

The project name can contain only alphanumeric characters and the underscore character.

2. If you want to save the files for the project to a different location, uncheck **Use default location** and specify a location for the project files.
3. Click **Finish** to create the project.

The new project is displayed in the project explorer.

You can now add Workbench objects such as datastores and replication jobs to your project.

4.3.2 Exporting resources and preferences

Use the Export wizard to export Workbench folders, resources, and user preferences.

1. Right-click within the project explorer pane and choose **Export**.

The Export wizard appears.

2. Select the type of export that you want to perform and click **Next**.

- Archive File

Exports selected Workbench resources to a .zip or .tar archive on the local file system.

- File System

Exports selected Workbench resources directly to a location on the local file system.

- Preferences

Exports Workbench configuration preferences to a location on the local file system.

3. If you are exporting to an archive file or a location on the file system, select the resources to export and set the export options as desired.

- a. Select the folders and resources that you want to export.

Folders appear in the left pane and resources appear in the right pane.

Tip:

Folders with a grayed-out selection box are not explicitly selected, but contain objects that will be exported.

- b. Click **Browse** to specify the location where the archive file or resource files will be created.
- c. If you are exporting to an archive file, select the type of archive file to create.
- d. Select the directory structure mode.

Option	Description
Create directory structure for files	Creates the directory structure for all selected resources, even if no folders are explicitly selected.
Create only selected directories	Creates the directory structure only for folders that are explicitly selected.

4. If you are exporting preferences, select the preferences to export and set the export options as desired.

- a. Select the preferences to export, or select **Export all** to export all Workbench preferences.
- b. Click **Browse** to specify the location where the preference file will be exported.

5. Click **Finish**.

The selected resources or preferences are exported to the specified archive file or filesystem location.

4.3.3 Importing resources and preferences

Use the Import wizard to import Workbench folders, resources, and user preferences.

1. Right-click within the project explorer pane and choose **Import**.
The Import wizard appears.
2. Select the type of export that you want to perform and click **Next**.
 - **Archive File**
Imports selected Workbench resources from a .zip or .tar archive on the local file system.
 - **File System**
Imports selected Workbench resource files from a location on the local file system.
 - **Preferences**
Imports Workbench configuration preferences from a file on the local file system.
3. If you are importing from an archive file or a location on the file system, select the resources to import and set the import options as desired.
 - a. Click **Browse** to specify the location of the archive file or resource files.
 - b. Select the folders and resources that you want to import.
Folders appear in the left pane and resources appear in the right pane.
Tip:
Folders with a grayed-out selection box are not explicitly selected, but contain objects that will be imported.
 - c. Click **Browse** to specify the folder where the imported resources will be stored.
4. If you are importing preferences, select the preferences to import.
 - a. Click **Browse** to specify the preference file to import.
 - b. Select the preferences to import, or select **Import all** to import all Workbench preferences.
5. Click **Finish**.
The selected resources or preferences are imported to the Workbench.

Index

\$LINK_DIR 10

D

- data
 - migrating 25
- data migration 25
- database migration 25
 - delta load 36
 - error recovery 35
 - replication behavior 30
- datastore editor 28
- delta load
 - application options 38
 - jobs 36
 - table options 37
- DS_COMMON_DIR 10
- DS_USER_DIR 10

E

- edit menu 17
- editor
 - datastore 28
 - replication job 29
- environment variables
 - \$LINK_DIR 10
 - DS_COMMON_DIR 10
 - DS_USER_DIR 10
- exporting
 - datastore 50
 - preferences 50
 - project 50
 - resources 50
- expression editor 40

F

- file menu 17
- functions 41

H

- help menu 19

I

- importing
 - datastore 51
 - preferences 51
 - project 51
 - resources 51
- INSTALL_DIR 10

L

- LINK_DIR 10
- logging in
 - Workbench 15

M

- migrating
 - data 25
 - schema information 25
- migration
 - data 25
 - database 25
 - schema 25
 - supported data sources 39
 - supported data targets 39

N

- naming conventions
 - BI platform 10
 - Information platform services 10
 - terminology 10
 - variables 10

R

- replication job editor 29
- replication wizard 26

S

- schema information
 - migrating 25
- schema migration 25
- system variables
 - DS_COMMON_DIR 10
 - DS_USER_DIR 10
 - LINK_DIR 10

T

- tools menu 17

U

- user interface 15, 16, 30, 31
 - datastore editor 22
 - edit menu 17
 - file menu 17
 - help menu 19
 - monitoring editor 23
 - project explorer 20
 - replication job editor 20
 - toolbar 19
 - tools menu 17
 - validate menu 18
 - window menu 19
 - workbench window 15

V

- validate menu 18

W

- window menu 19
- Workbench
 - logging in 15

