



Configuration Help

SAP Strategy Management 10.1

Target Audience

- Technical Consultants
- System Administrators

Document version: 1.1 – July 17, 2015

THE BEST-RUN BUSINESSES RUN SAP



Copyright

© 2015 SAP AG or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.






Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

National product specifications may vary.

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.

SAP 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. Please see <http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark> for additional trademark information and notices.

Icons in Body Text

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Additional icons are used in SAP documentation to help you identify different types of information at a glance. For more information, see *Help on Help → General Information Classes and Information Classes for Business Information Warehouse* on the first page of any version of *SAP Library*.

Typographic Conventions

Type Style	Description
<i>Example text</i>	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation.
Example text	Emphasized words or phrases in body text, graphic titles, and table titles.
EXAMPLE TEXT	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
Example text	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
Example text	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example text>	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
EXAMPLE TEXT	Keys on the keyboard, for example, F2 or ENTER.

Document History

Date	Description
October 1, 2013	Initial release
July 17, 2015	Added information about Strategy Management BI Connector Administrator (BICA) to the topic "Setting Up the SAP NetWeaver BW Connector".

Table of Contents

1	SAP Strategy Management Configuration.....	8
2	Planning	10
3	Implementing Your Strategy Management System.....	11
3.1	Setting Up the Demonstration	11
3.1.1	Building the HFPBM Model.....	12
3.1.2	Starting the Administration Application for the Demonstration	12
3.1.3	Creating a Model Connection	13
3.1.4	Assigning Roles to the Context.....	15
3.1.5	Assign the HFPBM Model Connection to the Fashions Enterprise Context	15
3.1.6	Touring the Application with the Fashions Enterprise Context	16
3.2	Getting Started with a Full Implementation.....	16
3.2.1	Model Development.....	17
3.2.1.1	Five Measures that Comprise a KPI.....	17
3.2.1.2	Overview of Source and Virtual Measures	18
3.2.1.3	Types of Scores	19
3.2.2	How Scores Color a KPI's Status Indicator	20
3.2.3	Model Creation	20
3.2.4	Creating Measures for the KPIs.....	20
3.2.5	Additional Information	22
4	Setting Up the SAP NetWeaver BW Connector	23
4.1	Creating a Link ID for SAP NetWeaver BW Connector	24
4.2	Using a Link ID.....	26
4.2.1	Troubleshooting Problems with the Link ID	26
4.3	Accessing Data from an SAP NetWeaver BW InfoCube	27
4.3.1	Application Server Database Creation.....	27
4.3.2	Mapping an InfoCube to an Application Server Database and Caching the Information	28
4.3.2.1	Creating an Application Server Database and Specifying the BEx Query Cube Name in the Schema Subsystem	30
4.3.2.2	Setting the Fiscal Year	30
4.3.2.3	Using the 0FISCPER Time Characteristic.....	32
4.3.2.4	Single-Step Scenario for Mapping Metadata	33
4.3.2.5	Multi-Step Scenario for Mapping Metadata	35
4.3.2.5.1	IMPORT DIMENSION	35
4.3.2.5.2	IMPORT TIME	36
4.3.2.5.3	IMPORT VARIABLES.....	37
4.3.2.5.4	IMPORT QUERY VARIABLES	38
4.3.3	Testing	38
4.3.4	Examining the Generated MDX.....	39

4.3.5 Caching in Application Server.....	40
4.3.6 Querying SAP NetWeaver BW Manually in ACCESS LSLINK.....	41
4.3.7 Related Application Server Commands	42
5 Integration with Other Systems.....	43
5.1 Setting a Link to the Application from Enterprise Portal	44
5.3 Providing Strategy Management Data for Other Systems.....	44
5.3.1 Adding SMDDataService Web Service Users to SAP NetWeaver UME	45
5.3.2 Setting Basic Authentication for the SMDDataService Web Service	45
5.3.3 API for the SMDDataService Web Service.....	46
5.4 Providing Application Server Data for Other SAP Systems	48
5.4.1 Adding CubeService Web Service Users to UME.....	49
5.4.2 Setting Basic Authentication for the CubeService Web Service Provider	49
5.4.3 API for CubeService Web Service	50
5.5 Providing Strategy Management Data for SAP BusinessObjects Dashboards.....	52
5.5.1 Exporting the WSDL files from SAP NetWeaver.....	53
5.5.2 Configuring Web Service Connections in SAP BusinessObjects Dashboard.....	54
5.5.3 Mapping Input and Output Values to an Excel Spreadsheet.....	55
5.5.4 Mapping SAP BusinessObjects Dashboard Components to the Excel Spreadsheet Data	56
5.6 Providing Strategy Management Data for WebI	57
5.6.1 Installing the Strategy Management ODBO Provider.....	58
5.6.2 Creating the Universe	59
5.6.3 Exporting the Universe	61
5.6.4 Creating a WebI Document and Generating a Report	61
5.7 Providing Strategy Management Data for Crystal Reports.....	61
5.7.1 Providing Data Using Query As A Web Service.....	63
5.7.1.1 Creating a QaaWS from a Universe	63
5.7.1.2 Creating a Query.....	63
5.7.1.3 Creating a Report Using QaaWS for Crystal Reports.....	64
5.7.2 Providing Data Using the Strategy Management ODBO Provider	65
5.7.2.1 Installing the Strategy Management ODBO Provider	65
5.7.2.2 Creating an OLAP Connection and Setting up a Report	67
5.8 Using an Application Server Fiscal Calendar for Queries in SAP BusinessObjects Enterprise	68
6 Application Server Configuration Files.....	69
6.1 LSSERVER.INI Settings	69
6.1.1 [Microsoft Windows] settings	69
6.1.2 [Server] Settings.....	72
6.1.3 [XXXXXX] Settings	72
6.2 LSDAL.INI Settings	74

6.2.1 [ConnectionID] Settings.....	74
6.2.2 [Enable] Settings	74
6.2.3 [Settings] Settings.....	74
6.3 LSDAL.CNF Settings	75



1 SAP Strategy Management Configuration

SAP Strategy Management is a comprehensive performance management software framework and includes software components for strategy and planning; initiative management and prioritization; scorecards; dashboards; and reports and ad hoc analysis.

This guide contains steps to configure the strategy management application. System administrators and technical consultants can use this guide to configure the following components that make up the application:

- **Application Server** stores the quantitative data for the scorecard, dashboard, and reports within Application Server. Application Server is a highly scalable, time-intelligent multidimensional OLAP engine. Application Server typically operates in a multidimensional mode (MOLAP), but can also operate in a relational OLAP mode (ROLAP) or a hybrid OLAP mode (HOLAP) as needed to satisfy the data requirements of the system.
- **Java Interactive Publisher (JPIP)** is the middle-tier components including authorization, business logic, data access, e-mail, data conversion, request workflow processing. The basic flow of the application is based on a model-view-controller architecture common to most standard Web applications.
- **Software Component Archive (SCA)** contains the database/dictionary for the SAP NetWeaver System database. The SCA is made up of SDAs, one of which contains all the middle tier logic that the applications require. Another SDA, `strategymanagementdic`, contains all the CPMS_ tables and table definitions. The tables have no data in them until you import `bootstrap.zip` or `import.zip`.

Integration

This configuration guide is related to the Installation Guide. Follow the procedures in this guide after you have installed the strategy management components.

For information about how other SAP systems are integrated with the strategy management application, see the *Master Guide for SAP Strategy Management* on the SAP Help Portal at <http://help.sap.com/bosm101>.

Constraints

This guide does not explain how to install or deploy the software. For information about installation, see the *Installation Guide for SAP Strategy Management* on the SAP Help Portal at <http://help.sap.com/bosm101>.

This guide does not explain how to install and configure the integrated components including SAP NetWeaver, SAP NetWeaver Portal, SAP NetWeaver BW, or SAP Scheduler. For information about these SAP installations, see the appropriate installation guides on the SAP Help Portal.

This guide does not explain how to configure the client machines of users who use the administration application and strategy management applications. For information about configuring client machines to run the administration application and the strategy management application, see the SAP Help Portal at <http://help.sap.com/bosm101>. Choose **Application Help** → **Startup Requirements** ←

This guide does not explain how to install or configure Excel Add-In. For information about this installation and configuration, see the *Client Installation Guide for SAP Strategy Management Excel Add-In* on the SAP Help Portal at <http://help.sap.com/bosm101>.

This guide does not explain how to download or configure the External Data Loader or Model Designer. For information, see the *SAP Strategy Management External Data Loader User's*

Guide and the *SAP Strategy Management Model Designer User's Guide* on the SAP Help Portal at <http://help.sap.com/bosm101>.



2 Planning

Use this process to configure your strategy management implementation.

Process

1. Begin your strategy management implementation.
For information, see [Implementing Your Strategy Management System](#) [Page 11].
2. Set up the SAP NetWeaver BW Connector if you intend to use it.
For information, see [Setting Up the SAP NetWeaver BW Connector](#) [Page 23].
3. Set up connections to other systems if you intend to use them.
For information, see [Integration with Other Systems](#) [Page 43].
4. Troubleshoot any startup issues.
For information, see [Application Server Configuration Files](#) [Page 69].



3 Implementing Your Strategy Management System

You can start working with the application in different ways.

Prerequisites

You are a strategy management administrator.

You have set up your browser. For more information, see the SAP Help Portal for SAP Strategy Management at <http://help.sap.com/bosm101>. Choose **Application Help** → **Startup Requirements** → **Required Software and Settings**.

System users and groups are populated in the strategy management system. For more information, see the *Server Installation Guide for SAP Strategy Management* on the SAP Help Portal at <http://help.sap.com/bosm101>.

Process

1. Set up and review the demonstration implementation.

You can view the definitions of a sample scorecard first before defining your own scorecard, review the demonstration files, and allow users to run the strategy management application with sample scorecard information.

For more information, see [Setting Up the Demonstration](#) [Page 11].

2. Set up your model and define a full scorecard implementation.

For more information, see [Getting Started with a Full Implementation](#) [Page 16].



3.1 Setting Up the Demonstration

You can view the definitions of a sample scorecard first before defining your own scorecard, and allow users to use the strategy management application with sample scorecard information.

Prerequisites

The administrator of SAP NetWeaver has created roles.

You are the strategy management administrator.

There are different types of administrative permissions for each topic. If one user is performing all steps, that user should be the strategy management administrator to have access to all features. If various users are carrying out the steps, they should have the exact permissions set in the individual Prerequisites sections of the related topics.

Process

1. In Application Server, build the `HFPBM` model by running a procedure file.
2. Start the Launch page, and click the administration application link associated with the `pwsample`.

The `pwsample` option contains sample data.

3. Create a model connection.

4. Assign the model connection to the *Fashions Enterprise* context.
5. Assign roles to the context.
6. Tour the application with the *Fashions Enterprise* context.



3.1.1 Building the HFPBM Model

You build the HFPBM sample model so that you can make the database available to the application.

Prerequisites

You are the administrator of Application Server.

Procedure

Procedure

If Application Server is installed on a Microsoft Windows server

1. Start Application Server on the Microsoft Windows server where it is installed.
2. Issue the command:

```
JOB HFPBMMAK.PRO;ext
```

If Application Server is installed on a Microsoft Windows client and accessing Application Server on a Linux/UNIX server in a client/server configuration

1. Copy `HFPBM.DMP` and `HFPBMMAK.PRO` from the Microsoft Windows client to the `$DBHOME` directory on the UNIX/Linux server.



Case is significant on Linux/UNIX. The dump file and procedure file must be capitalized.

2. Start Application Server on the Microsoft Windows client.
3. Issue the command:

```
JOB HFPBMMAK.PRO;ext
```



3.1.2 Starting the Administration Application for the Demonstration

After you add the sample model to Application Server's `MASTERDB`, you can start working in the administration application.

Prerequisites

You are the strategy management administrator.

The global properties in SAP NetWeaver Administrator are modified to use `pwsample` for the `PWDatabase` property.

The HFPBM model is added to Application Server's `MASTERDB`.

Procedure

1. Open a browser window and type the following to start the administration application:

```
http://<nw_server>:<port>/strategy
```
2. Depending on the authentication set up at your site, you may be prompted to log on. If you are prompted, log on with your `pipadmin` administrator username and password.
3. In the Launch page, select the administration application link associated with `pwsample` data.



3.1.3 Creating a Model Connection

After you add the sample model to Application Server's `MASTERDB`, you can start working in the administration application to create a model connection.

You define the model connection that associates your users to an Application Server username and then to an Application Server model. This model connection is then later associated with the Fashions Enterprise context.

Prerequisites

The HFPBM model is added to Application Server's `MASTERDB`.

You are running the administration application (associated with the `pwsample` data) as the strategy management administrator.

Procedure

1. In the administration application, click **Administration** → **Manage Models**.



If **Administration** is unavailable to you, it means you do not have the correct permissions.

2. Select the **New** link at the bottom of the **Model Connections** list. The **Model Connection** text box appears in the **Connection Settings** section.
3. In the **Model Connection** text box, type the following model connection name:

```
HFPBM
```
4. In the **Web Server Name** text box, type the name of the server where Application Server is running. If Interactive Publisher is installed on a different server than SAP NetWeaver, then enter the IP address for Application Server.
5. In the **Web Server User** text box, type the authentication name of a user on the Application Server system. On Microsoft Windows, this user must be a member of the administrators group. On Linux/UNIX, this user must have permission to run the scripts and programs in the Application Server installation directory.
6. In the **Password** text box, type the password for the authentication user.
7. In the **PAS Model** text box, type the following:

```
HFPBM
```



The model connection must be named HFPBM, the same name as the model name, to view the scorecards in the demonstration model.

8. Use these default settings:

PAS User – GUEST

Password – leave an empty text box

Port – 8325

INI file – lsserver.ini

Service – PILOT

Min Instances – 0

Max Instances – 5



Make sure the Application Server user is a user of the database you are adding to the model connection definition.

9. Click *Test connection*.

A message displays, stating that the Application Server connection is made, you are logged in, and the Application Server *USE* database was found.

10. By default, the *Everyone* option is selected in the *Groups and Users* section and all users are added to the model connection when you create it. Do any of the following to add users to this model connection:
 - o Keep the default setting of allowing all users (*Everyone*) to use this model connection. You can click *Save* and skip the rest of the steps in this topic. Only one Application Server user can be assigned the *Everyone* setting for a model connection. If you attempt to create a second Application Server user with the *Everyone* setting, you a message telling you to pick specific users or groups.
 - o In the *Groups and Users* section, choose specific system groups or users for this model connection.

Click *System Groups* to assign system groups. If your site has populated users lists from an LDAP/ActiveDirectory server, then this list is populated with lists of system groups to select. Otherwise, this list is empty and this option is unavailable.

Click *Users* to assign individual users.

From the left-hand list, do any of the following to add users and system groups:

- Select a user or system group and click *Add* to add it to the right-hand list.
- Drag the cursor over several users and system groups, and click *Add* to add multiple members.

- Press CTRL and then click the users to select them, and then click *Add* to add nonadjacent users.

11. Click *Save*.



3.1.4 Assigning Roles to the Context

Now you assign the roles to the *Fashions Enterprise* context. This gives the roles access to the context.

Prerequisites

The HFPBM model is added to Application Server's `MASTERDB`.

You are a context administrator (a user in a role with permission to create contexts). If

► *Context Management* → *Manage Contexts* ◀ is unavailable, it means you do not have the correct permissions.

The model connection is created.

You have created the Roles *SSM_Executive* and *SSM_Services* following the instructions in Chapter 7 of the *SAP Strategy Management Installation Guide*. Users are assigned to these roles and all the strategy management *Create* and *View* actions are assigned to these roles.

Procedure

1. In the administration application, select ► *Context Management* → *Manage Contexts* ◀.
2. Select the *Fashions Enterprise* context.
3. In the *Roles* list, select *SSM_Executive* and *SSM_Service*.
4. Save your changes.



3.1.5 Assign the HFPBM Model Connection to the Fashions Enterprise Context

Now you will associate the *Fashions Enterprise* context with the *HFPBM* model connection. This step connects the scorecard with the model. This is an important step because this connection allows you to view the sample scorecard definitions in the administration application and also in the application.

Prerequisites

You are a strategy management administrator or a scorecard administrator (any user assigned to a role with permission to create scorecards). If ► *Context Management* → *Manage Scorecards* ◀ is unavailable, it means you do not have the correct permissions.

Procedure

1. In the administration application, select ► *Context Management* → *Manage Scorecards* → *Scorecard Defaults* ◀.
2. From the *Context* list, select *Fashions Enterprise*.
3. From the *PAS Model Connection* dropdown list, select *HFPBM*.
4. Save your changes.



3.1.6 Touring the Application with the Fashions Enterprise Context

Users assigned to the *SSM_Executive* or *SSM_Service* role can now start the application, select the Fashions Enterprise context and display the available tabs to become familiar with the application. Users can also access the administration application if they have been granted access.

Procedure

1. Open a browser window and type the following to start the application:

```
http://<nw_server>:<port>/strategy
```

2. Depending on the authentication set up at your site, you may be prompted to log on. If you are prompted, log on with your Web authentication username and password.
3. Select the strategy management application link associated with the *pwsample database*.



3.2 Getting Started with a Full Implementation

These steps show the general flow of implementing a scorecard beginning with the development of the Application Server model through rolling out the context to users in the application.

Prerequisites

There are different types of administrative permissions for each topic. If one user is performing all steps, that user should be the strategy management administrator to have access to all features. If various users are carrying out the steps, they should have the exact permissions set in the individual Prerequisites sections of the related topics.

Process

1. Develop the Application Server model.

For more information about developing Application Server models, see the online Help in the Application Server Administrator program.

1. Determine the appropriate formulas to create the score, trend, and gap performance measures.
 2. Understand how scores color a KPI's status indicator.
 3. Create measures for the KPIs.
2. Set up system defaults, model connections, and set up the library.
 1. Start the administration application.
 2. Set application defaults and system defaults.
 3. Create a model connection.
 4. Create a context, assign the model connection to the context, and assign roles to the context.
 5. Create the perspectives and objectives and KPIs in the library.

3. Assign perspectives, objectives, indexed KPIs, and KPIs to the context.
4. Define the perspectives, objectives, and KPIs within the context.
5. Set up a strategy, which includes the goal diagram, themes or pathways, and causes and effects.
6. Access the strategy management application to create initiatives, and review the scorecard and strategy.

Result

In the application, any user with access to the *Initiatives* tab, *Scorecard* tab, and *Strategy* tab can work with the initiatives, objectives, and strategy you defined.



3.2.1 Model Development

As in all standard models, you need to identify the dimension structures and their hierarchies, the attributes and measures, and how the measures are dimensioned. In addition, identify the fiscal calendar, the periodicity of the measures, and the period for which to load data.

This topic assumes that you already have a model with dimensions and measures containing loaded data. For more information about creating a model, see the online Help in the Application Server. There are KPIs and indexed KPIs:

- Indexed KPIs are composed of KPIs and their measurement is derived from the measurements of the KPIs. For information, see the Administrator Help in SAP Library.
- KPIs are composed of five measures from the Application Server model. This topic explains how to create KPIs.

The first step toward implementing scorecards is to create an Application Server model to create the ratio-based measures used by the KPIs. You use the Application Server program to review your measures and determine the calculations for the measures you want to look at.

This section describes information about creating the measures in a model that are the basis for KPIs. This section does not provide a full explanation of developing the entire model. For information about building a model, see the online Help in the Application Server program.



3.2.1.1 Five Measures that Comprise a KPI

For every KPI that you want to create for a scorecard, you must create five measures in the Application Server model.



You do not set up measures for index KPIs, which are simply composed of KPIs.

This list shows the measures you need to create for a single KPI. You create two measures from source data, and three measures are virtual measures.

Measure	Description	Example
Actual	Contains source data from actual numbers.	Sales-Actual
Target	The target or budget measure stored in your model from source data.	Sales-Target

Measure	Description	Example
Score	This is a virtual measure calculated from the actual and target measures. For example, Sales-Score might represent actual sales as a percentage of target sales.	Sales-Score
Trend of Actual	This is a virtual measure calculated from the moving average of the actual measure.	Sales-Trend
Gap Performance	This is a virtual measure calculated as current period's score in comparison to the moving average score over time. It indicates the score's performance over time.	Sales-Gap Performance



3.2.1.2 Overview of Source and Virtual Measures

There are typically three types of measures in a model: source, calculated, and virtual. The models used in a strategy management implementation use source and virtual measures.

Source measures

Source measures contain data that is loaded into the model from a source file, typically a relational database or flat file. Common source measures could include:

- Sales
- Cost
- Margin
- Price
- Units
- Commissions
- Quota
- Headcount
- Overhead

In most standard models, there is typically a Type dimension that has actual and budget or plan members. In a model used for strategy management scorecards, there must be actual and target data and these data points must be created as measures rather than dimension members.

Source measures used in a Scorecard implementation

You must create two source measures, actual and target, for every Key Performance Indicator (KPI) you want to use in a scorecard:

- **Actual** — The actual measure stored in your model from source data.
- **Target** — The target or budget measure stored in your model from source data.

Virtual measures

Virtual measures are calculations whose data values are generated dynamically when you request them, and they are not stored permanently.

Virtual measures used in a Scorecard implementation

In a scorecard implementation, the score, trend of actual, and gap performance measures are virtual measures for a KPI.

- **Score** — Calculated as needed from the actual and target measures.
- **Trend of Actual** — Calculated as needed from the moving average of the actual measure.
- **Gap Performance** — Calculated as needed to measure the current period's score in comparison to the moving average score over time. It indicates the score's performance over time.



3.2.1.3 Types of Scores

The following table shows the types of scores you might use:

Target	Purpose	Example KPIs	Optimal Score	Formula
Achievement	For scenarios where you want actual to exceed the target	Revenue, Profit, # Cases shipped, # Customers	Actual greater than target	Actual as a percent of target or (Actual % Target)
Reduction	For scenarios where you want actual to be less than the target	Cost, Expense, Overtime	Actual less than target	$(100 - ((\text{Actual} - \text{Target}) / \text{Target}))$
Absolute	For scenarios where you want actual to always equal target	Inventory	Actual equal to target	$(100 - \text{ABS}((\text{Actual} - \text{Target}) / \text{Target}))$
Zero	For situations when you want actual and target to be 0 (or close to it)	Product defects, Employee sick days, Employee attrition	Actual and target close to 0	(Actual - Target)
Deviation	For scenarios where you want to know the gap between actual and target	N/A	N/A	$((\text{Actual} - \text{Target}) / \text{Target})$



3.2.2 How Scores Color a KPI's Status Indicator

The application uses the *Score* value for a KPI to determine the color of that KPI's status indicator. The *Score* values are matched to user-defined index values that are matched to different status colors. The color associated with the index value that matches the score becomes the color of the KPI's status indicator.

There are several steps to building a KPI:

1. Determining the appropriate formulas to create the score, trend, and gap performance measures. This step is performed in Application Server.
2. In the administration application, you set up the KPI for the scorecard by selecting its five measures from the Application Server model, and you determine the appropriate index values to color the status indicator. Which index values you use depends on the formula you used to create the KPI's score measure, and the results you expect.

When these two steps are completed, the user can quickly see the KPI's measures, the association between the score measure and the index values, and how and why the KPI was given a particular status color.



3.2.3 Model Creation

As in all standard models, identify the dimension structures and their hierarchies, the attributes and measures, and how the measures are dimensioned. In addition, identify the fiscal calendar, the periodicity of the measures, and the period for which to load the data.

This topic assumes that you already have a model with dimensions and measures containing loaded data. For more information about creating a model, see the online Help in the Application Server program.



If you are using attributes in your model, make sure the short names do not contain plus signs (+). If they do have plus signs, users receive an error message when selecting those members in the *Dimensional Selector* dialog box in the Reports component.



3.2.4 Creating Measures for the KPIs

Follow these steps for every measure you want to develop into a KPI for your scorecard implementation.

You must create the measure names using the following format:

Measure Type	Measure Name
Actual	<measure>_ACT
Target	<measure>_TAR
Score	<measure>_TARDEV
Trend of Actual	<measure>_TRD
Gap Performance	<measure>_TRDDEV



If you are implementing a Type dimension, only the _TAR measures should be dimensioned by the Type dimension.

Procedure

1. Determine the dimensions in the model for which the measures are to be dimensioned.
2. Start the Application Server program.
3. Create the actual measure.

This shows an example of a Sales measure that is associated with the Customer, Dealer, and Vehicle dimensions:

```
CREATE MONTHLY VARIABLE Sales_ACT LABEL 'Sales-Actual' BY  
Customer, Dealer, Vehicle
```

4. Create the target measure. For example:

```
CREATE MONTHLY VARIABLE Sales_TAR LABEL 'Sales-Target' BY  
Customer, Dealer, Vehicle
```

5. Create the Score measure using one of these calculations below. The type of calculation depends on the type of values you are measuring.
 - o **Achievement percent scores** account for scenarios where you want actual to exceed the target.

Example of KPIs with achievement targets: Revenue, Profit, # Cases shipped, # Customers

The calculation for an achievement percent score is as follows:

```
CREATE VARIABLE Sales_TARDEV LABEL 'Sales-Score' BY  
Customer, Dealer, Vehicle AS (Sales_ACT % Sales_TAR)
```

- o **Reduction percent scores** account for scenarios where you want actual to be less than the target

Example of KPIs with reduction targets: Cost, Expense, Overtime

The calculation for a reduction percent score is as follows:

```
CREATE VARIABLE Sales_TARDEV LABEL 'Sales-Score' BY  
Customer, Dealer, Vehicle AS (100 - ((Sales_ACT -  
Sales_TAR) % Sales_TAR))
```

- o **Absolute percent scores** account for scenarios where you want actual to always equal target.

Example of KPIs with absolute targets: Inventory

The calculation for an absolute percent score is as follows:

```
CREATE VARIABLE Sales_TARDEV LABEL 'Sales-Score' BY  
Customer, Dealer, Vehicle AS (100 - ABS((Sales_ACT -  
Sales_TAR) % Sales_TAR))
```

- o **Zero scores** account for situations when you want actual and target to be 0 or close to it.

Example of KPIs with zero targets: Product defects, Employee sick days, Employee attrition

The calculation for zero score is as follows:

```
CREATE VARIABLE Sales_TARDEV LABEL 'Sales-Score' BY
Customer, Dealer, Vehicle AS ((Sales_ACT - Sales_TAR)
```

- o **Deviation percent scores**

The calculation for a deviation percent score is as follows:

```
CREATE VARIABLE Sales_TARDEV LABEL 'Sales-Score' BY
Customer, Dealer, Vehicle AS ((Sales_ACT -
Sales_TAR)%Sales_TAR)
```

6. In cases where actual values do not exist for some periods but target values do exist, you can make an adjustment to the formula to prevent misleading score values. The adjustment involves multiplying the formula by $(\text{actual}+.001)/(\text{actual}+,001)$. With this adjustment in place, if the actual value is missing, then the score value is missing. For example, the CREATE VARIABLE command for the reduction percent score is adjusted as follows:

7. CREATE VARIABLE Sales_TARDEV LABEL 'Sales-Score' BY Customer, Dealer, Vehicle AS $(100 - ((\text{Sales_ACT} - \text{Sales_TAR})\% \text{Sales_TAR})) * ((\text{Sales_ACT} + .001) / (\text{Sales_ACT} + .001))$

8. This adjustment is not necessary for achievement percent calculations.

9. Create the Trend of Actual measure. For example:

```
CREATE VARIABLE Sales_TRD LABEL 'Sales-Trend of Actual' BY
Customer, Dealer, Vehicle AS MOVING2 (Sales_ACT,1,3)
```

10. Create the Gap Performance measure using one of these formulas:

Calculation for achievement percent, reduction percent, absolute percent, or deviation percent gap performance:

```
CREATE VARIABLE Sales_TRDDEV LABEL 'Sales-Gap Performance' BY
Customer, Dealer, Vehicle AS (Sales_TARDEV-
MOVING2(Sales_TARDEV,1,3))%ABS(MOVING2(Sales_TARDEV,1,3))
```

Calculation for a zero gap performance:

```
CREATE VARIABLE Sales_TRDDEV LABEL 'Sales-Gap Performance' BY
Customer, Dealer, Vehicle AS (Sales_TARDEV-
MOVING2(Sales_TARDEV,1,3))%MOVING2(Sales_TARDEV,1,3)
```



3.2.5 Additional Information

For information about all the strategy management administrator tasks available in the administration application, see the SAP Help Portal for SAP Strategy Management at <http://help.sap.com/bosm101>. Choose **Application Help** → **Administration**.



4 Setting Up the SAP NetWeaver BW Connector

You can set up the application to use SAP NetWeaver BW data as a source for your scorecard and reporting implementation. There are two scenarios for implementing SAP NetWeaver BW Connector:

- The scorecard data remains in SAP NetWeaver BW and is read as and when required.
- All of the relevant data is brought from SAP NetWeaver BW into Application Server and cached. This scenario is important where a large volume of scorecard data must be integrated.

Scorecard databases can be created from BEx Query Cubes (preferred) or InfoCubes. There are cases when you must use BEx Query Cubes to work with Application Server. You must use a Query Cube if:

- You want to access Navigational Attributes in NetWeaver BW from Application Server. Raw InfoCubes in the \$InfoCube catalog do not expose Navigational Attributes. If you want to use them in Application Server, you must create a BEx Query cube.
- The underlying InfoCube has too many Characteristics.

This section describes how to create a Link ID that is used to connect to SAP Netweaver BW. It also explains the Application Server commands that are used to connect to and retrieve data from SAP NetWeaver BW.

Notes:

- The Strategy Management BI Connector Administrator (BICA) simplifies the process of creating Application Server databases sourced from SAP Netweaver BW. BICA is a browser-based tool that can be used to connect to BW and import relevant data from selected queries. See SAP Note 2131423 for information about configuring BICA.
- SAP NetWeaver BW is case sensitive, while Application Server is not. In SAP NetWeaver BW, characteristics called Material and MATERIAL are different objects, while in Application Server they are not. The code that caches metadata from InfoCubes in Application Server must on occasion modify names for use in Application Server, though always using the NetWeaver BW name when querying SAP NetWeaver BW. If you execute your own MDX, you must make it case sensitive.

Prerequisites

You are running a minimum of SAP NetWeaver BI 7.0 min SP15 (SAP_BASIS SP13) or SAP NetWeaver BI 3.5 min SP20.

The NetWeaver BW backend is running on a platform supported by NetWeaver BW.

You have a basic understanding of Application Server.

You can log on to Application Server as the administrator.

Process

1. Create a Link ID in Application Server Administrator.
2. Use the Link ID in Application Server Administrator.
3. Do one of the following:

- If the amount of data is small, you can create an Application Server database that contains SAP cube data at the input level. Then use the application with the Application Server database that contains the InfoCube data.
- Map the InfoCubes to the Application Server database and cache the information about the InfoCubes in the Application Server database. Then use the application with the Application Server database that is connected with the InfoCubes.

4.1 Creating a Link ID for SAP NetWeaver BW Connector

In the application, a component called *Link* acts like DB Connect in SAP NetWeaver BW. It provides a means to access external data sources like RDBMS (through ODBC or Oracle OCI or Db2 CLI) and BAPI in a seamless, consistent way.

A Link ID stores all the relevant logon and connection setting information to connect to an external data source. Once you create a Link ID, an Application Server application can access that external data source just using the Link ID name alone. Information required to connect to the external data source using that Link ID is stored in `lsdal.ini` (in the Microsoft Windows directory on Microsoft Windows or by default in your `$LSSHOME` directory on UNIX).

Prerequisites

You have Application Server installed in one of these configurations:

- On a Microsoft Windows server
- On a Linux/UNIX server. with a client copy installed on a Microsoft Windows machine

In the steps below, when the term **Microsoft Windows machine** is used, it means either Microsoft Windows server or Microsoft Windows client, depending on your implementation noted above.

Procedure

1. Start Application Server Administrator on the Microsoft Windows machine. Go to **Start** → **Programs** → **Application Server Administrator**.
2. Choose **File** → **New** → **Link ID** to display the *Create Link ID* dialog box.
3. Select *SAP NW BW RFC* and click *OK*. RFC supports connections through load balancing or a specific server.
4. In the *Link ID Properties* dialog box, enter the values for RFC. This dialog box shows different fields depending on the type of RFC you selected.



All the entries in the fields of the *Link ID* dialog box correspond exactly to those fields that you see in the *SAP Logon* pad in the *Connection* tab of the dialog. You must enter the same case sensitive values that you would enter in the *SAP Logon* pad *Connection* tab.

Fields for SAP System Related to Load Balancing	Description

Fields for SAP System Related to Load Balancing	Description
<i>System</i>	Name of SAP System of the BW back-end server. This corresponds to the entry <i>System ID</i> on the <i>Connection</i> tab of the <i>SAP Logon</i> pad.
<i>Message Server</i>	Host name of message server.
<i>Group</i>	Optional. Group name of the application servers. Default setting is <i>PUBLIC</i> .
<i>UserId</i>	SAP NetWeaver BW user who has access to the data in the cube/query. This user can be a dialog user (someone who can run front-end SAP GUI applications) or a communication user (someone who can connect and access data). This user must be able to run any query and retrieve all the results for that query.
<i>Password</i>	Password of SAP NetWeaver BW user.
<i>Language</i>	Client language
<i>Client</i>	Client Number. This is back-end-specific, which your back end administrator can provide.
<i>Rfc Trace (0 or 1)</i>	1 is on and 0 is off. The default is 0.
Fields for SAP System Using a Specific Server	Description
<i>System</i>	Name of the SAP System of the BW back-end server. This corresponds to the entry <i>System ID</i> on the <i>Connection</i> tab of the <i>SAP Logon</i> pad.
<i>Application Server</i>	Host name of a specific SAP application server.
<i>SAP System number</i>	SAP system number.
<i>UserId</i>	SAP NetWeaver BW user who has access to the data in the cube/query. This user can be a dialog user (someone who can run front-end SAP GUI applications) or a communication user (someone who can connect and access data). This must be able to run any query and retrieve all the results for that query.
<i>Password</i>	Password of the SAP NetWeaver BW user.
<i>Gateway Host</i>	Optional. Default is the gateway on the application server.
<i>Gateway Server</i>	Optional. Default is the gateway on the application server.
<i>Language</i>	Client language.
<i>Client</i>	Client Number. This is back-end-specific, which your back end administrator can provide.
<i>Rfc Trace (0 or 1)</i>	1 is on and 0 is off. The default is 0.

5. Click *OK*.

6. In Application Server Administrator, click the *Link IDs* entry and then click the *List* tab on the main window. The new Link ID is displayed.
7. (Additional step for Linux/UNIX server implementations running a client version of Microsoft Windows) Transfer the `lsdal.ini` file from the Microsoft Windows client directory to the UNIX server directory where you installed Application Server. Ensure the filename `lsdal.ini` is in lower case on the Linux/UNIX system. Make sure the environment variable `LSLINKINI` points to the directory containing the `lsdal.ini` file in order for Application Server to find it.

When the Link ID is saved, the system stores a copy of the connection information in `\Windows\lsdal.ini`. Passwords are encrypted in `lsdal.ini`.

4.2 Using a Link ID

Once you create a LinkId, you can use it in the `ACCESS LSLINK` or `SCHEMA` subsystems to access the SAP NetWeaver BW backend. `ACCESS LSLINK` allows you to enter manual MDX statements and see the results by issuing a `PEEK` statement.

The `SCHEMA` subsystem allows you to query BI metadata about cubes, characteristics, hierarchies, levels, members, and properties through BAPI. Metadata is not provided through MDX (MDX is only for queries). BAPI returns metadata through Schema Rowsets, which are tabular data sets.

The easiest way to check that your LinkId is working correctly is to display a list of all the available InfoCubes and Query Cubes on the BI back end associated with the LinkId. In the *Command* window in Application Server Administrator, issue the following:

```
SCHEMA

System>schema

CONNECT <link-id>

VIEW cubes ROWSET

END
```

You see a two-column, tab-separated output with a list of the Info cubes available and their cube type. The cube type is either a raw info cube, which is type `CUBE`, or a `QUERY CUBE`, with the correct `$` prefix for raw cubes and the correct `<parent info cube>/<technical query name>` for Query Cubes.

The `VIEW <rowset type> ROWSET` command is useful for examining BW metadata. All the output is in tab-separated form, which you can copy to Excel to take a look at it. Internally, whenever you issue any `IMPORT` commands against an InfoCube, the `IMPORT` command is using these rowsets to query BW metadata and cache it in an Application Server database in Application Server metadata.

4.2.1 Troubleshooting Problems with the Link ID

When creating a Link ID, if you encounter problems, there are some tests you can perform to troubleshoot the issue.

Procedure

- First, see if you can access the target system using the *SAP Logon* pad. If you cannot access it this way, then contact the system administrator to resolve this.

- If you can access the target system through the *SAP Logon* pad, then check that all the entries in the LinkId correspond exactly with the corresponding entries in the *SAP Logon* pad. Remember that these entries are case sensitive.

On UNIX, the shared libraries are found by the loader according to the shell variables `LD_LIBRARY_PATH` (Sun, HP64 bit, Linux) or `SHLIB_PATH` (HP 32 bit) or `LIB_PATH` (AIX). For information about shared libraries, see the *Installation Guide for SAP Strategy Management* on the SAP Help Portal at <http://help.sap.com/bosm101>. This guide discusses the setting of these variables for the Unicode ICU libraries.



4.3 Accessing Data from an SAP NetWeaver BW InfoCube

There are two ways to use SAP NetWeaver BW data in Application Server:

- If the amount of InfoCube data is small, you can create an Application Server database that contains the InfoCube data at the input level. In the application, you can select the Application Server database that contains the InfoCube data.
- Map the InfoCubes to the Application Server database and cache the information about the InfoCubes in the Application Server database. In the application, you can select the Application Server database that is connected with the InfoCubes.



4.3.1 Application Server Database Creation

You can create an Application Server database from an SAP NetWeaver BW InfoCube. You import the InfoCube input data and then consolidate it in Application Server and use the Rollup editor features. Using this method, you never have to access the InfoCube at runtime.

`IMPORT DATA` is a Hybrid OLAP `SCHEMA` subsystem command that imports data defined in SAP NetWeaver BW into an existing Application Server database.

Internally, `IMPORT DATA` imports data for one variable at a time and generates an MDX statement for each one. `IMPORT DATA` only imports `INPUT` data (transaction data `IMPORT DATA` fetches up to 1 million cells in any one MDX query). If any single variable hits the 1 million limit barrier, the data for that variable is not imported. Importing InfoCube data directly into an Application Server database is best used on small cubes. We recommend this method for quick display times.

If you want to display SAP NetWeaver BW data from larger InfoCubes, you must use the `IMPORT DIMENSION`, `IMPORT TIME`, `IMPORT VARIABLES`, and `IMPORT QUERY VARIABLES` commands to access the InfoCube data without actually importing it into Application Server. By accessing the data without importing it, you retrieve and cache information about the cubes in the Application Server database while the data remains in the InfoCube.

Example

```
SCHEMA

IMPORT SCHEMA

SELECT DIM2 INPUT

SELECT DIM3 INPUT

IMPORT DATA SELECTED
```

```

SET VARIABLE * NOFROM (sets a native measure in Application Server)

ROLLUP SALES

END

ADD EVERY

END

SET PERIOD DEFAULT

CONSOLIDATE SALES

CONSOLIDATE COSTS

CONSOLIDATE . . . .

```



4.3.2 Mapping an InfoCube to an Application Server Database and Caching the Information

You can map one or more InfoCubes into an Application Server database using Hybrid OLAP to cache BW metadata inside an Application Server database in Application Server metadata format.

When you map InfoCubes to an Application Server database, the **Key Figures**, **Characteristics**, and **Navigational Attributes** from the InfoCubes appear in the Application Server database as normal Application Server measures, dimensions and attributes. No data is actually copied into Application Server. At runtime, Hybrid OLAP generates the appropriate MDX statements to query the corresponding InfoCubes. The resulting view is the same as if the data was actually in Application Server. The Application Server database appears to any application just like any other Application Server database. The application does not know that the data for the view came from SAP NetWeaver BW, and so does not require any changes in the application.

The Application Server `IMPORT` commands retrieve and cache the BW metadata in an Application Server database. The `IMPORT` commands are the same as in standard Hybrid OLAP except that there are none of the schema tables that we have in HOLAP against a relational database management system (RDBMS). All the schema information comes from the OLAP BAPI rowsets.

For information about the `IMPORT` commands, see the online Help in the Application Server Administrator program.

Prerequisites

You have created a functional Link ID.

Process

1. Create an Application Server database to act as the cache for the BW metadata. Then enter the `Schema` subsystem and connect to your target BW Link ID. Then specify the name of the InfoCube that contains the metadata to retrieve.
2. Do one of the following to map the BEx Query Cube into Application Server:
 - If you created a BEx Query Cube that maps neatly into Application Server (meaning one that has fewer than 12 Characteristics excluding Time and Navigational Attributes, and has only 1 Time Characteristic), issue an `IMPORT SCHEMA` command.

- If the BEx Query Cube does not map neatly, issue `IMPORT DIMENSION`, then `IMPORT TIME`, then `IMPORT VARIABLES` commands.

4.3.2.1 Creating an Application Server Database and Specifying the BEx Query Cube Name in the Schema Subsystem

Procedure

1. Create an Application Server database to act as the cache for the BW metadata. Issue the command:

```
SUPER CREATE DATABASE <database> BLOCKS 10000 OBSERVATIONS 1000  
USE <database>
```

2. If the start month of your fiscal calendar is not January, you must issue the `SET FISCAL CALENDAR` command. Then you must create a document in Application Server that matches the first period in the BW query/Infocube to a fiscal period in Application Server.

For information, see [Setting the Fiscal Year](#) [Page 30].

If your query uses the `0FISCPER` time characteristic, the document must be created even for a January start month.

For information, see [Using the 0FISCPER Time Characteristic](#) [Page 32].

3. Enter the Schema subsystem and connect to your target BW LinkId:

```
SCHEMA
```

```
System>schema
```

```
CONNECT <link-id>
```

4. Specify the name of the Cube that contains the metadata to retrieve. For example, enter this depending on whether it is a BEx Query Cube or a raw InfoCube:

```
CUBE '0D_SD_C03/ZTEST_MAT_HIER_2'
```

or

```
CUBE '$0D_SD_C03'
```

5. Use an `IMPORT` command to map the BEx Query Cube to the Application Server database.

More Information

[Single-Step Scenario for Mapping Metadata](#) [Page 33]

[Multi-Step Scenario for Mapping Metadata](#) [Page 35]

4.3.2.2 Setting the Fiscal Year

If the start month of your fiscal calendar is not January, you must issue the `SET FISCAL CALENDAR` command in Application Server to define the correct start month. This must be done prior to importing any variables into the model.

Application Server reads a subset of the time characteristics available in BW. Only one time characteristic per query/Infocube is read and additional time values are calculated on-the-fly

by Application Server. For example, if the `0CALMONTH` time characteristic is imported, then quarterly and yearly values are calculated.

When Application Server interprets the time values from BW, it uses the fiscal year setting in the model. For example, the value `201101` in a BW query with the `0CALMONTH` time characteristic is associated with April 2011 if your fiscal year begins in April.

You can control the interpretation of the BW time values by creating document sets in the Application Server model. The name of the document set depends on the time characteristic of your BW query. If you import multiple queries with the same time characteristic, they are all interpreted in the same way. If you import multiple queries with different time characteristics, you need one document set for each time characteristic.

The table below explains the document sets required for the various BW time characteristics. Create the document sets in your Application Server model and enter one of the lines from the *Example Contents* column. Application Server uses the one line of information to determine how to interpret all time values read from BW. These examples use 2011 as the fiscal year but any year can be used.

BW Time Characteristic	Document Set Name	Example Contents	Explanation
0CALYER	BWFISCYRINFO	2011 2011 FISCAL	BW time values are defined based on the end month of the fiscal year. In BW, YYYY is the year associated with the last month of the fiscal year.
		2011 2010 FISCAL	BW time values are defined based on the start month of the fiscal year. In BW, YYYY is the year associated with the first month of the fiscal year.
0CALQUARTER	BWFISCQTRINFO	20111 20111	BW time values are defined based on the start month of the fiscal year. YYYY1 is the first quarter of the year that starts in YYYY
		FISCAL20111 20101	BW time values are defined based on the end month of the fiscal year. YYYY1 is the first quarter of the year that ends in YYYY
		FISCAL20111 20111 CALENDAR	BW time values are defined based on the calendar year. YYYY1 is always January - March.
0CALMONTH	BWFISCMONINFO	201101 201101	BW time values align with the fiscal year start month. YYYY01 is the first month of the fiscal year
		FISCAL201101 201101 CALENDAR	BW time values align with the calendar months. YYYY01 is always January.



If you use the SAP BW Connector Administrator (BICA) to create your Application Server models, you are prompted for the information required to create these document sets when the fiscal year start is not January. The only exception is for the 0CALMONTH time characteristic. BICA makes the assumption that the BW time values align with calendar months and always creates BWFISCMONINFO with 201101 201101 CALENDAR.

4.3.2.3 Using the 0FISCPER Time Characteristic

For most of the SAP NetWeaver BW time characteristics, Application Server can map the time values to a calendar date in Application Server when the fiscal year start month is January. However, for the 0FISCPER time characteristic, the time information is available in the format YYYYPPP. PPP represents a period number and could be a day, month, or quarter. You must create a document set in Application Server to match the format with a calendar date regardless of the fiscal year start month.

The steps below assume that you are importing a BW query with a 0FISCPER time characteristic and that your fiscal year starts in October.

Procedure

1. Create a document set in Application Server called BWFISCPERINFO and add these two lines:

```
<periodicity>
<YYYYPPP><YYYYMM[DD]>
```

Variable	Value
<periodicity>	Application Server periodicity represented in the 0FISCPER time characteristic. For example, MONTHLY or QUARTERLY. Specify a minimum of 3 characters.
<YYYYPPP>	BW representation of the first period of a fiscal year. YYYY is the fiscal year. PPP is the period of that fiscal year. For example, 2009001 represents the first period in fiscal year 2009.
<YYYYMM[DD]>	Calendar equivalent of YYYYPP .

The following two lines in the BWFISCPERINFO document indicate that the data is monthly and 2009001 in BW corresponds to the calendar date of October 2008 in Application Server. The 2009 fiscal year range is from October 2008 through September 2009.

```
MONTHLY
2009001 200810
```

2. Issue a SET FISCAL CALENDAR command in Application Server before importing a BW query. For example, if you are using the document set described above, the command would be SET FISCAL CALENDAR OCTOBER.



4.3.2.4 Single-Step Scenario for Mapping Metadata

If you created a BEx Query Cube that maps equal items into Application Server, you can issue an `IMPORT SCHEMA` command to map all the metadata into Application Server in one simple step.

If the following two items are true, follow the steps in this section:

- The source InfoCube or Query Cube has at most 12 Characteristics (excluding Time and Navigational Attributes).
- There is only one Time Characteristic.



Only monthly, quarterly, and yearly periodicities are supported for loading data from SAP NetWeaver BW to Application Server.

Prerequisites

You have logged on to Application Server Administrator as the administrator, created a database for caching purposes, accessed the Schema subsystem, and specified the name of the BEx Query Cube or InfoCube.

Features

In the Schema subsystem of Application Server Administrator, issue the following command:

```
IMPORT SCHEMA [INCLUDING FISCAL][RANGE <date1>-<date2>][SPANS
{DERIVED|EXACT}][HIERARCHIES {ALL|DEFAULT}] [SYSVAR] [CHARACTERISTIC
{'OCALMONTH'|'OCALQUARTER'|'OCALYEAR'}][FORCE]
```

Activities

Internally, the `IMPORT SCHEMA` command does the equivalent of:

```
IMPORT FISCAL
```

```
IMPORT DIMENSION
```

```
IMPORT TIME
```

```
IMPORT VARIABLES
```

```
IMPORT QUERY VARIABLES (from BEx Query cubes)
```

See the [Hybrid OLAP Help → SCHEMA Subsystem Command Reference → IMPORT SCHEMA](#) (for SAP NetWeaver BW Connector) in the online Help in Application Server Administrator for details about this command.

Example

In this example, there is a Link ID called Bilinkid, which defines connectivity information about the BW backend Q52. The Query Cube is called 'OD_SD_C03/ZTEST_MAT_HIER_2'. The Query Cube has at most 12 Characteristics, has only one Time Characteristic (OCALMONTH) and only sensible Time Characteristic Values. The Application Server database is called Mytestdb.

You would use the following commands to map that cube into an Application Server database:

```

USE juice EXCLUSIVE

SUPERVISOR CREATE DATABASE Mytestdb BLOCKS 10000

USE Mytestdb EXCLUSIVE

SCHEMA

SCHEMA>CONNECT Bilinkid

SCHEMA>CUBE 'OD_SD_C03/ZTEST_MAT_HIER_2'

SCHEMA>IMPORT SCHEMA

All 9 Members of 0MATERIAL;MYTESTDB Selected

All 68 Members of GOODS_RECIPIENT;MYTESTDB Selected

All 19 Members of SALES_ORGANIZATION;MYTESTDB Selected

All 15 Members of SALES_GROUP;MYTESTDB Selected

All 112 Members of REGION;MYTESTDB Selected

8 Members Roll Into Multiple Outputs

All 22 Members of OD_MATERIAL;MYTESTDB Selected

All 5 Members of DIVISION;MYTESTDB Selected

All 6 Members of DISTRIBUTION_CHANNEL;MYTESTDB Selected

All 2 Members of COUNTRY;MYTESTDB Selected

All 8 Members of COMPANY_CODE;MYTESTDB Selected

Creating Attribute COUNTRY

1 Variable Created

1 Variable Created

1 Variable Created

1 Variable Created

1 Variable Created

1 Variable Created

1 Variable Created

1 Variable Created

1 Variable Created

1 Variable Created

1 Variable Created

SCHEMA>End

```

In Application Server Administrator, you can refresh the *Dimensional Models* pane, and expand the MyTESTDB database to see all the Dimensions and Attributes in the database. You can click the *Measures* item and then click the *List* tab in the main window to see a list of the measures and all their properties.



4.3.2.5 Multi-Step Scenario for Mapping Metadata

If you created a BEx Query Cube that does not map equally into Application Server, you must issue individual `IMPORT DIMENSION`, `IMPORT MEASURES`, and `IMPORT TIME` commands to map all the metadata into Application Server.

If one or both of these items are true, follow the steps in this section:

- The source InfoCube or Query Cube has more than 12 Characteristics (excluding Time and Navigational Attributes).
- There is more than one Time Characteristic.



Only monthly, quarterly, and yearly periodicities are supported for loading data from SAP NetWeaver BW to Application Server.



4.3.2.5.1 IMPORT DIMENSION

The `IMPORT DIMENSION` command retrieves the metadata for BW characteristics from the target cube and creates corresponding dimensions in Application Server.

You issue the commands to import just the dimensions you want to use for analysis from the cube. Since in OLAP BAPI the Navigational Attributes appear as Characteristics, you import Navigational Attributes into Application Server Attributes this way too.

Prerequisites

You have logged on to Application Server Administrator as the administrator, created a database for caching purposes, accessed the Schema subsystem, and specified the name of the BEx Query Cube or InfoCube.

To import dimensions that exist in several different BW queries, the dimensions must be identical; they must have the same filters, restrictions, members, query variables, default hierarchy, and so on.

Features

If the raw InfoCube or Query Cube has at most 12 characteristics, excluding Time and Navigational Attributes, you can issue the `IMPORT DIMENSION *` command to import all of them. If you use the `IMPORT DIMENSION *` command then the `IMPORT` automatically figures out what Navigational Attributes there are to import.

```
IMPORT DIMENSION *
```

If you have a raw InfoCube that has more than 12 Characteristics, you issue individual `IMPORT DIMENSION` commands to import just the dimensions you want to use for analysis from the cube (and `IMPORT` at most 12 of those). Since in OLAP BAPI the Navigational Attributes appear as Characteristics, you import Navigational Attributes into Application Server Attributes this way too. If you import them one at a time, you should import the Characteristic first and the related Navigational Attributes afterwards.

```
IMPORT DIMENSION { * | { <dimension> [,<dimension>...] } } [INCLUDING FISCAL] [RANGE <date1>-<date>>] [HIERARCHIES {ALL|DEFAULT}] [FORCE ]
```

If the raw InfoCube or Query Cube has more than 12 characteristics, then issue individual `IMPORT DIMENSION` commands to import just the dimensions you want to use for analysis.

Use the BW technical dimension name enclosed in square brackets and the put names in single quotes. You can import up to 12 characteristics this way:

```
IMPORT DIMENSION '[ <name> ] '  
IMPORT DIMENSION '[ <name> ] '  
...
```

The Application Server dimension name is based on the BW Characteristic description. For example, if a BW Characteristic name is `OD_DIS_CHAN` and has the description "Distribution channel", the Application Server dimension name is `DISTRIBUTION_CHANNEL`.

The dimension hierarchies correspond to equivalent BW hierarchies. The hierarchy names are based on the hierarchy captions in SAP NetWeaver BW.

Level names in SAP NetWeaver BW have no names. They are simply Level00, Level01, Level02, and so on. The dimension levels use the names Level00, Level01, and so on. The Total member of the dimension is at Level00, and its children are at Level01.

All the Characteristic Values from the BW Characteristic are imported as members in the Application Server dimension.

The member short name is based on the BW short name (`MEMBER_NAME`). The member long name is based on the BW long name (`MEMBER_CAPTION`).

See the [Hybrid OLAP Help → SCHEMA Subsystem Command Reference → IMPORT DIMENSION](#) (for SAP NetWeaver BW Connector) in the online Help in Application Server Administrator for details about this command.



4.3.2.5.2 IMPORT TIME

There are 13 Time Characteristics in SAP NetWeaver BW. The `IMPORT TIME` command identifies which Time Characteristic to use. Application Server only uses one characteristic and must be able to derive a date from it.

The `IMPORT TIME` command is a unique `IMPORT` command in the `Schema` subsystem used only for SAP NetWeaver BW Connections.

Use `IMPORT TIME` to import the BW Time dimension that contains Time Characteristics. This allows Application Server to determine a calendar date in the fact records on an external data source that we can use to map into an internal Julian date.

Prerequisites

You have issued the `IMPORT DIMENSION` command in the `Schema` subsystem.

You are in the `Schema` subsystem in Application Server Administrator.

Features

If you have created a Query Cube that has only one Time Characteristic, then you can issue this command:

```
IMPORT TIME
```

If the BW Time dimension has more than one Time characteristic, issue the following command with one of these Time characteristics. Do not use any other Time characteristics:

```
IMPORT TIME [INCLUDING FISCAL] [RANGE <date1>-<date2>] [SPANS
{DERIVED|EXACT}] [SYSVAR] [CHARACTERISTIC {'0CALMONTH'|
'0CALQUARTER'|'0CALYEAR'}] [FORCE]
```

Use 0CALMONTH for monthly data, 0CALQUARTER for quarterly data, and 0CALYEAR for yearly data.

For information, see the *Application Server Help* on the SAP Help Portal at <http://help.sap.com/bosm101>. Then choose **Hybrid OLAP Help** → **SCHEMA Subsystem Command Reference** → **IMPORT TIME** (for SAP NetWeaver BW Connector).



4.3.2.5.3 IMPORT VARIABLES

Use the `IMPORT VARIABLE` command to import information about Key Figures from SAP NetWeaver BW into Application Server variables (measures). The information is derived from the OLAP BAPI Measures rowset. The `IMPORT VARIABLE` command determines the dimensionality and data type (INTEGRAL BYTES 1/2/4 or NUMERIC BYTES 4/8) and creates the variable with as much information as is available from the SAP NetWeaver BW.

Then use the `SET VARIABLE` command to set properties like `RATE`, `EXPENSE`, `UNITS`, `DECIMALS`, `WIDTH` and whether the variable is to be time converted with `SUM`, `FIRST`, `LAST`, and so on.

The `IMPORT VARIABLES` command creates variables dimensioned by all the nonattribute dimensions you have `IMPORTED` from the corresponding InfoCube. A measure is dimensioned by all the dimensions imported from the corresponding InfoCube. There is no mixed dimensionality within a single InfoCube.

If measures are dimensioned differently then they would have to be created in different InfoCubes and either linked as `MULTICUBE`, or handled as separate cubes. You can have an Application Server database into which you have `IMPORTED` dimensions and variables from more than one cube.

If you have two InfoCubes, CubeA and CubeB, then to map both into a single Application Server database you do the following:

```
CONNECT BiTest

CUBE CubeA

IMP SCHEMA

CUBE CubeB

IMP SCHEMA
```

Prerequisites

You have issued the `IMPORT TIME` command in the `Schema` subsystem.

You are currently in the `Schema` subsystem in Application Server Administrator.

Features



Issue the following command:

```
IMPORT VARIABLES { * | <variable> [,<variable>...] } [FORCE]
```

A BEx Query Cube may have Calculated Key Figures. These are derived measures that have no data stored permanently in the InfoCube, but are calculated at runtime. When you import Calculated Key Figures to Application Server, they are treated no differently than other BW

Key Figures. At runtime they are imported together. There is no way for you to determine from within Application Server whether a measure is a loaded or calculated measure in SAP NetWeaver BW, or determine what the calculation formula is.

If you define virtual variables in Application Server they work as normal. The Virtual Variable only exists in Application Server. At runtime, the system fetches the appropriate data from the InfoCubes for all base measures used in any Virtual Variables. The Virtual Variable calculations are performed in Application Server at runtime in the same way that they are calculated in native Application Server. This means that you can create a Virtual Variable in Application Server that performs a calculation based on Key Figures from different InfoCubes.

See the  [Hybrid OLAP Help →SCHEMA Subsystem Command Reference →IMPORT VARIABLES](#)  (for SAP NetWeaver BW Connector) in the online Help in Application Server Administrator for details about this command.



4.3.2.5.4 IMPORT QUERY VARIABLES

The `IMPORT QUERY VARIABLES` command imports any query variable information from the `SAP VARIABLES` rowset for a Query Cube and caches it in Application Server.

You can examine the cache using the `EXHIBIT QUERY VARIABLES` command and you can set values for a Query Variable using the `SET QUERY VARIABLE` command.

Query Variables are placeholders in the Query that values can be supplied at runtime, rather than hard coding selections or values in the Query itself.

Query Variables can be placeholders for member names in a dimension (a characteristic value in a characteristic or, a dimension hierarchy. Query variables can also be placeholders for a numeric value that is used in a constraint or numeric value used in a formula in a calculated measure. Query Variables can be optional or mandatory and may or may not have a default value.

At runtime, SAP NetWeaver has implemented extensions to MDX so that an application can supply values for Query Variables to the back end. The BW Connector takes any values that you have set using the `SET QUERY VARIABLE` command and incorporates them into the generated MDX to pass them to the backend.



Prerequisites

You are currently in the `Schema` subsystem in Application Server Administrator.

Features

Issue the following command:

```
IMPORT QUERY VARIABLES [FORCE]
```

See the  [Hybrid OLAP Help →SCHEMA Subsystem Command Reference →IMPORT QUERY VARIABLES](#)  (for SAP NetWeaver BW Connector) in the online Help in Application Server Administrator for details about this command.



4.3.3 Testing

Once you have all the information cached in Application Server, you can quickly and easily try some quick ad hoc navigation and querying.

Procedure

1. Click the *Data View* tab in the Application Server window to view an ad hoc navigation grid.
2. Click inside the view to display the *Viewer* dialog box.
3. Drag dimensions and attributes to the *Across* and *Down* and *Page* areas to specify what you want to view. You can double-click any dimension, attribute, or measures or Time in the *Viewer* to display the *Dimensional Selector* dialog box.
4. Double-click to move members (characteristic values) into and out of the view.
5. When you do double-click *Time*, the *Calendar* dialog box is displayed. If there are multiple hierarchies, you can switch hierarchies with the *Hierarchy* dropdown and you can select members from multiple hierarchies if you want.
6. When you are done, Application Server fetches all the required data from SAP NetWeaver BW and displays it in the grid. Within the grid you can drill up and down on members by double clicking. This shows you that the InfoCube is being accessed correctly by Application Server.



4.3.4 Examining the Generated MDX

You can examine the generated MDX used to access SAP NetWeaver BW. You do so by issuing commands in Application Server.

Prerequisites

You should be familiar with the `SELECT`, `SET PERIOD` and `LIST` commands in Application Server. For information, see the Application Server Help on the SAP Help Portal at <http://help.sap.com/bosm101>.

Features

Use the `Schema` command `SPY {ON | OFF | external text file name}` to start echoing the generated MDX.

If you issue commands such as these:

```
SET Mon PERIOD 2009/1/1-2010/12/31

SELECT VARIABLES 'COST STATS CURRENCY'

SELECT OD_MATERIAL
L4_DAMENKLEIDUNG, '1004', CN0F21, CNSERVICE, HERRENKLEIDUNG

ACROSS TIME Down VARIABLES, OD_MATERIAL

SELECT DIMENSION COMPANY_CODE ALL

SELECT DIMENSION COUNTRY ALL

SET LONG

LIST
```

The output appears as follows:

```
System> list

SELECT {[Measures].[1HIXHAC44YID6QLF23WLYRQ01]} ON 0,
```

```

NON EMPTY {[0CALMONTH].[200901]:[0CALMONTH].[201012]} DIMENSION
PROPERTIES MEMBER_UNIQUE_NAME ON 1,
NON EMPTY {[0D_MATERIAL MATERIAL].[CN0F21],[0D_MATERIAL
MATERIAL].[CNSERVICE],[0D_MATERIAL MATERIAL].[1004
0D_MTLGROUP],[0D_MATERIAL MATERIAL].[HERRENKLEIDUNG
OHIER_NODE],[0D_MATERIAL MATERIAL].[DAMENKLEIDUNG
OHIER_NODE]} DIMENSION PROPERTIES MEMBER_UNIQUE_NAME ON 2
FROM [0D_SD_C03/ZTEST_MAT_HIER_2
WHERE
([0MATERIAL].[All],[0D_CO_CODE].[All],[0D_DIS_CHAN].[All],[0D_DIV].[A
ll],
[0D_SHIPTO].[All],[0D_REGION].[All],[0D_SALE_GRP].[All],[0D_SALE_ORG]
.[All])
Jan 09 Feb 09 Mar 09 Apr 09
Cost stats currency
CN0F21 33338.283K 16317.999K 34206.339K 28870.065K
Pulli 33338.283K 16317.999K 34206.339K 28870.065K
HERRENKLEIDUNG 230617.46K 136451.35K 182923.60K 115826.56K
DAMENKLEIDUNG 373771.70K 277218.67K 338042.79K 264889.01K
Then you can issue LIST QUARTERLY, LIST YEARLY, or LIST WEEKLY to see the time
converted.

```



4.3.5 Caching in Application Server

Application Server caches information from SAP NetWeaver BW to improve performance. Some of the information is cached permanently within the Application Server database, some is cached at the start of a new session and some is cached temporarily.

The dimensions in Application Server are cached permanently. If the Characteristics or Navigational Attributes change in SAP NetWeaver BW, you should issue new `IMPORT` commands in Application Server to display up-to-date dimensions in Application Server.

The `TIME` characteristic information is cached permanently by `IMPORT TIME`. After that, in any Application Server session, the Connector examines the timestamps in the `CUBES` rowset by default every 10 minutes to see if any new data has been loaded into the InfoCube. If it has, then the Time Cache is updated. If the `USE` database is open for writing, the permanent cache is updated. If the `USE` database is open only for reading, then the system creates a new temporary cache for that session on the `Work` database. The system queries the Time Characteristic Values and creates an internal lookup of Characteristic Values to internal Julian dates. This is used throughout the session to convert the Time Characteristic Values returned in MDX queries to internal Julian dates. If more data is added to an InfoCube (for example, for a new month), you do not have to do anything in Application Server. A new session picks up all the correct dates. If new data has been load and the permanent cache

could not be updated because the `USE` database was only opened for reading, a new import should be done at the next convenient time to update the permanent cache.



Features

Use the `CACHE TIME <seconds>` command to control how often the Connector checks to see if new data has been loaded. The default is 600 seconds (10 minutes).

In Application Server, the measure definition is cached permanently, but the time span (date range) for a measure is cached in the same way as the Time Characteristics — attempts are made to update it regularly. Every Application Server session should have up to date information about Time.

SAP Query Variables information is cached permanently.

Views are cached temporarily as described above.

See the  *Hybrid OLAP Help → SCHEMA Subsystem Command Reference → CACHE (for SAP NetWeaver BW Connector)*  in the online Help in Application Server Administrator for details about this command.



4.3.6 Querying SAP NetWeaver BW Manually in ACCESS LSLINK

You can query BW directly in `ACCESS LSLINK` just like any other external source. You use MDX rather than SQL.

Prerequisites

You should be familiar with MDX.

Features

The query returns a multi dimensional cross tabulation containing the dimensions and members you have on your axes in your MDX query. `ACCESS LSLINK` can only handle two dimensional results sets.

For example, you might have only two axes in your MDX query and only one dimension on each axis. Application Server flattens the rowset for BW within `Access LSLink`.

The following commands:

```
ACCESS LSLINK

CONNECT BITEST

SELECT {[Measures].[1HIXHAC44YID6QLF23WLYRQ01]} ON 0,
NON EMPTY {[0CALMONTH].[200301]} DIMENSION PROPERTIES
MEMBER_UNIQUE_NAME, MEMBER_CAPTION ON 1,
NON EMPTY {[0D_MATERIAL].[CN0F21], [0D_MATERIAL].[CNSERVICE]}
DIMENSION PROPERTIES MEMBER_UNIQUE_NAME, MEMBER_CAPTION ON 2

FROM [0D_SD_C03/ZTEST_MAT_HIER_2]

PEEK
```

produce results such as:

```
ACCESS PILOT LINK> PEEK TABS NONUM
```

```
[OD_MATERIAL].[LEVEL01].[MEMBER_UNIQUE_NAME]
[OD_MATERIAL].[LEVEL01].[MEMBER_CAPTION]

[OCALMONTH].[LEVEL01].[MEMBER_UNIQUE_NAME]
[OCALMONTH].[LEVEL01].[MEMBER_CAPTION]

[Measures].[1HIXHAC44YID6QLF23WLYRQ01]

[OD_MATERIAL].[CN0F21] CN0F21 [OCALMONTH].[200301] JAN 2003
33338283.00
```



4.3.7 Related Application Server Commands

The following commands are available for SAP NetWeaver BW Connector.

VIEW SPANS TYPE

VIEW TIME RANGE

VIEW <rowset> ROWSET

EXHIBIT QUERY VARIABLES

EXHIBIT DIMENSION <dimension> MDXNAME

EXHIBIT MEASURE MDXNAME

EXHIBIT VARIABLE MDXNAME

SET QUERY VARIABLES

See the [Hybrid OLAP Help](#) → [SCHEMA Subsystem Command Reference](#) in the online Help in Application Server Administrator for details about this command.



5 Integration with Other Systems

The strategy management application has interfaces to other systems:

Prerequisites

You are the strategy management administrator.

Enterprise Portal is installed and running if you want to set up a link to the strategy management application from Enterprise Portal.

Features

You can set the following technical interfaces:

- Set up a link from the Enterprise Portal to the strategy management application.

For information, see [Setting a Link to the Application from Enterprise Portal](#) [Page 44].

- Configure the strategy management application to access SAP Business Planning and Consolidation data. You can create initiatives and KPIs using Planning and Consolidation data and use them in your scorecard.

For more information, see the SAP Help Portal at <http://help.sap.com> → *Analytics* → *All Products*. Then filter on *Financial Information Management*. Choose *SAP BusinessObjects Financial Information Management 10.0: Supplement for SAP Strategy Management 10.0*.

- Allow other SAP systems to extract strategy management application data for their own purposes.

For more information, see [Providing Strategy Management Application Data for Other Systems](#) [Page 44].

- Allow other SAP systems to extract Application Server data for their own purposes.

For more information, see [Providing Application Server Data for Other SAP Systems](#) [Page 48].

- Allow users to create SAP BusinessObjects Dashboards using strategy management data.

For more information, see [Providing Strategy Management Data for SAP BusinessObjects Dashboards](#) [Page 52]

- Allow SAP BusinessObjects Analysis, edition for OLAP users to display strategy management data in SAP BusinessObjects Analysis.

For more information, see [Providing Strategy Management Data for SAP BusinessObjects Analysis](#) [Page 57]

- Allow WebI users to display strategy management data using WebI.

For more information, see [Providing Strategy Management Data for WebI](#) [Page 57]

- Allow SAP Crystal Report users to create reports in SAP Crystal Reports using strategy management data.

For more information, see [Providing Strategy Management Data for SAP Crystal Reports](#) [Page 61]



5.1 Setting a Link to the Application from Enterprise Portal

You can add a link to the strategy management application from the Enterprise Portal. When a user clicks the link, the strategy management application launches in a new browser window.

Prerequisites

You have Enterprise Portal and you are an administrator of Enterprise Portal.

Procedure

1. Start the Enterprise Portal Administrator.
2. In the Portal Content, create an iView.
3. In the URL text box, type the following:

```
http://<nw_server>:<port>/strategy/pilotworks/eplaunch.htm
```

4. Set a *GET Request Method*.



5.3 Providing Strategy Management Data for Other Systems

You can use the *SMDDataService* Web Service API to extract data from the strategy management application to another SAP system. Using the *SMDDataService* Web Service API, other systems can obtain performance management scorecard information including contexts, perspectives, KPIs, index KPIs, initiatives, and objectives.

Process

1. Add *SMDDataService* Web Service users to SAP NetWeaver UME.

For information, see [Adding SMDDataService Web Service Users to SAP NetWeaver UME](#) [Page 45].

2. Set *Basic Authentication* for the *SMDDataService* Web Service.

For information, see [Setting Basic Authentication for the SMDDataService Web Service](#) [Page 45].

3. Use the API for the *SMDDataService* Web Service to acquire strategy management contexts, perspectives, KPIs, index KPIs, initiatives, and objectives for use in another system.

For information, see [API for the SMDDataService Web Service](#) [Page 46]

5.3.1 Adding SMDataService Web Service Users to SAP NetWeaver UME

Any user who uses the *SMDataService* Web Service must be added to SAP NetWeaver UME. This is because basic authentication of the *SMDataService* Web service is executed by SAP NetWeaver UME.

Prerequisites

You are an administrator of SAP NetWeaver.

Procedure

Add Web Service users for *SMDataService* to SAP NetWeaver UME. For information, see the *Installation Guide for SAP Strategy Management* on the SAP Help Portal at <http://help.sap.com/bosm101>. Choose *Adding Strategy Management Users to SAP NetWeaver UME*.



5.3.2 Setting Basic Authentication for the SMDataService Web Service

You must configure the *SMDataService* Web Service provider to use Basic Authentication.

Prerequisites

You are the administrator of SAP NetWeaver.

Procedure

1. Start *Web Services Administration* in SAP NetWeaver using the following URL:
`http://<nw_server>:<port>/nwa`
2. Choose  *SOA* → *Application and Scenario Communication* → *Single Service Administration* .
3. Select the *Service Definitions* tab if it is not currently selected.
4. In the *Find* text box, enter *iSMDataService* and click *Go*.
5. In the *WSDL Port Type Name* list, select *iSMDataService* if it is not already selected.
6. Select the *Configuration* tab.
7. From the *Service Endpoint* list, select *SMDataServicePort*.
8. Click *Edit*.
9. Select the *Security* tab.
10. Do one of the following

If using SAP NetWeaver UME, in the *HTTP Authentication* section, select *User ID/Password* and then click *Save*.

If using SAP BusinessObjects authentication, in the *HTTP Authentication* section, make sure *User ID/Password* is not selected, and then click *Save*.



5.3.3 API for the SMDataService Web Service

The *SMDataService* Web Service is deployed in SAP NetWeaver AS Java. You access the strategy management Web Service Description Language (WSDL) using the following URL in a browser window:

```
http://<nw_server>:<port>/strategyServer/SMDataServiceService?wsdl&mode=sap_wsd1
```

The Web service authentication is handled by SAP NetWeaver and is set to Basic authentication and requires a username and password.

Prerequisites

SMDataService Web Service users are added to SAP NetWeaver UME.

Basic Authentication is set for this Web Service Provider in SAP NetWeaver Administrator.

Features

The following table shows the list of supported Web Service Methods for performance management data.

Web Service Method Name	Description
AllContextByDBName	Returns a list of all contexts for a given database name.
AllDBName	Returns a list of all dbnames in the strategy management application.
AllInitiativesByDBName	Returns a list of all Initiatives for a given context name and database name.
AllKpisByDBName	Returns a list of all KPIs for a given context name and database name.
AllPerspectiveByDBName	Returns a list of all perspectives for a given context name and database name.
AllScorecardByDBName	Returns a list of all scorecards for a given database name.
createInitiative	Creates an initiative in the application.
createTasks	Creates a milestone for the initiative in the application.
getAllContext	Returns a list of all contexts for a given database id.
getAllinitiatives	Returns list of all Initiatives for given context id and database id
getAllKPIs(long contextId, long dbNameId)	Returns a list of all KPIs for a given context id and database id
getAllPerspective	Returns a list of all perspectives for a given context id and database id.

Web Service Method Name	Description
getAllScorecard	Returns a list of all scorecards for a given database id.
getComment	Returns comments in the application for KPIs, objectives, and initiatives.
getIds	Returns all objectives with assigned KPIs or without assigned KPIs. If an ID is missing, -1 is returned.
getInitiativesById	Returns details of an initiative for a given Initiative id.
getInitiativesByObjectiveId	Returns the initiatives supporting an objective.
getKpiById	Returns a KPI corresponding to a specific id
getKpisByInitiative	Returns a list of KPIs for a given initiative id.
getKpisByObjective	Returns list of KPIs for a given objective id
getObjectiveById	Returns details of the objective for a given objective id
getObjectivesByInitiative	Returns a list of objectives for a given initiative id.
getObjectivesByPerspectiveId	Returns a list of objectives for a given perspective id.
getObjectiveStatus	Returns the objective status.
getPerspectiveById	Returns the perspective details for a given perspective id.
getTasksById	Returns details of the task for a given task id.
getTasksByInitiative	Returns a list of tasks for a given initiative id.
InitiativeByName	Returns details of an initiative for a given initiative name.
KPIByName	Returns a KPI corresponding to a specific name.

The schema for all the response values is detailed in a WSDL document. All the methods that return lists of objects populate the minimum identifying information of the objects (which includes just ID, Name, and Description). However, all the pertinent information is present when an object is retrieved by ID.

You can see a sample located at:

http://<nw_server>:<port>/strategy/tools/stratbrowser.jsp



5.4 Providing Application Server Data for Other SAP Systems

You can use the *CubeService* Web Service API to extract data from the strategy management application to another SAP system.

The *CubeService* Web Service is deployed in SAP NetWeaver. You access the strategy management Web Service Description Language (WSDL) using the following URL in a browser window:

```
http://<nw_server>:<port>/strategy/CubeServiceService?wsdl=sap_wsd1
```

The Web service authentication is handled by SAP NetWeaver and is set to Basic authentication and requires a username and password.

SAP Systems can obtain Application Server data including dimensions, members, hierarchies, measures, and Time. For more information, see [API for CubeService Web Service](#) [Page 50].

Process

1. Add users of the *CubeService* Web Service to SAP NetWeaver UME.

For information, see [Adding CubeService Web Service Users to UME](#) [Page 48].

2. Set Basic Authentication for the *CubeService* Web Service.

For information, see [Setting Basic Authentication for the CubeService Web Service](#) [Page 49].

3. Use the API for the *CubeService* Web Service.

For information, see [API for CubeService Web Service](#) [Page 50].



5.4.1 Adding CubeService Web Service Users to UME

Any user who uses the *CubeService* Web Service must be added to SAP NetWeaver UME. This is because basic authentication of the *CubeService* Web service is executed by SAP NetWeaver UME.

Prerequisites

You are an administrator of SAP NetWeaver.

Features

You must add Web Service users for *CubeService* to SAP NetWeaver UME. For information, see the *Installation Guide for SAP Strategy Management* on the SAP Help Portal at <http://help.sap.com/bosm101>. Choose *Adding Strategy Management Users to SAP NetWeaver UME*.



5.4.2 Setting Basic Authentication for the CubeService Web Service Provider

You must configure the *CubeService* Web Service provider to use *Basic Authentication*.

Prerequisites

Users planning to use the API for *CubeService* are known users in SAP NetWeaver UME.

You are the administrator of SAP NetWeaver.

Procedure

1. Start *Web Services Administration* in SAP NetWeaver using the following URL:
`http://<nwce_server>:<port>/nwa`
2. Choose **SOA** → *Application and Scenario Communication* → *Single Service Administration*.
3. Select the *Service Definitions* tab if it is not currently selected.
4. In the *Find* text box, enter `CubeServiceInterface` and click *Go*.
5. In the *WSDL Port Type Name* list, select *CubeServiceInterface* if it is not already selected.
6. Select the *Configuration* tab.
7. From the *Service Endpoint* list, select *CubeServicePort*.
8. Click *Edit*.
9. Select the *Security* tab.
10. Do one of the following
If using SAP NetWeaver UME, in the *HTTP Authentication* section, select *User ID/Password* and then click *Save*.

If using SAP BusinessObjects authentication, in the *HTTP Authentication* section, make sure *User ID/Password* is not selected, and then click *Save*.



5.4.3 API for CubeService Web Service

The CubeService Web Service is deployed in SAP NetWeaver. You access the strategy management Web Service Description Language (WSDL) using the following URL in a browser window:

`http://<nw_server>:<port>/strategyServer/CubeService?wsdl&mode=sap_wsdl`

The Web service authentication is handled by SAP NetWeaver and is set to Basic authentication and requires a username and password.

The CubeService Web service uses XMLA concepts of **discover** and **execute** to obtain information from the Application Server data.

Features

You can do the following using the Web Service Methods:

- Retrieve information from Application Server. In XMLA, the discover method is used to obtain metadata information about the OLAP cube. The strategy management Web service uses the Cubedef class to obtain metadata information.
- Execute queries/commands to Application Server using the executeQuery() method. Commands are written in DQL, which is the Application Server propriety language. The execute method allows applications to run provider-specific commands against XMLA data sources.

The following table shows the list of supported Web Service Methods for Application Server data.

Web Service Method Name	Description
executePASQuery	Returns the output of the Application Server command.
getAllInitiativeStatus	Returns the status information of all initiatives.
getDimensionDetails	Returns a dimension object for the specified dimension and model
getDimensionHierarchiesCount	Returns the number of hierarchies for the specified dimension and model.
getDimensionHierarchiesNames	Returns hierarchy names for the specified dimension and model.
getDimensionLevels	Returns the levels for the specified dimension and model
getDimensionNames	Returns a list of all dimensions for the specified model
getDimensions	Returns a list of dimension objects for the specified model
getDimMembers	Returns a list of member objects for the specified dimension and hierarchy and node at the specified

Web Service Method Name	Description
	<p>start number.</p> <p>Count is the number of results that should be displayed.</p> <p>Sort specifies the sort order. It can be default, ascending or descending.</p>
getDimMembersCount	Returns the number of members for the specified dimension and hierarchy at the specified node.
getDimMembersList	Returns a list of member objects for the specified dimension and hierarchy at the specified node. The hierarchy is the hierarchy name. The default hierarchy is default.
getInitiativeStatus	Returns the status of the given Initiative id and TimeInfo (periodicity and asOfDate)
getKPIStatus	Returns the status, which includes the actual, gap, score, status, target, trend, and trend status information of a given KPI by id and TimeInfo (periodicity and asOfDate)
getKPIStatusByTime	<p>Returns the status, which includes the actual, gap, score, status, target, trend, and trend status information of a given KPI by id and TimeInfo range (periodicity and asOfDate)</p> <p>Specify asOfDate as values separated by comma (.). For example,asOfDate - Jan 2011, Feb 2011 returns status for 2 months for the given KPI.</p>
getMeasureDetails	Returns the measure object for the specified measure name and cube.
getMeasureNames	Returns names of all measures for the specified cubename
getMeasures	Returns a list of all measure objects for the specified cubename
getModels	Returns the model connections.
getPeriod	Returns the time period for the specified cube.
getTimeEarliest	Returns the earliest time for the specified cube.
getTimeLatest	Returns the latest time for specified cube.

To execute query statements on Application Server, use the following method:

```
executeQuery(OLAPProperties, Command)
```

The method returns the output as a RowSet object. The output result is NOT in cellset format. The webservice client would have to parse the result string to obtain cellset or other formats as needed.

OLAPProperties includes the datasource name and language attributes.

Command includes the type attribute, which would be dql (or some other MDX language in future) and the Statement attribute, which is the actual query statements in dql (or some other MDX language).

ExecuteQuery() is stateless and if multiple statements/commands are required to be executed, then the client should pass ALL the statements/commands in single SOAP packet with each statement enclosed in square brackets [].

For example:

```
<OLAPPproperties>
<DataSourceInfo>HFPBM</DataSourceInfo>
<Locale>en</Locale>
</OLAPPproperties>
<Command type="dql">
<Statement>
[set dimension customer hierarchy default][exhibit dimension customer
only just below default_customer]
</Statement>
<Command>
<RowSet>
<Row> The output of the command.... </Row>
</RowSet>
```



5.5 Providing Strategy Management Data for SAP BusinessObjects Dashboards

You can make strategy management data available to users of SAP BusinessObjects Dashboards, allowing them to create Dashboards with the data.

The strategy management application provides the *CubeService* and *SMDDataService* Web Services. The *CubeService* Web Service provides the ability to acquire Application Server data and the *SMDDataService* Web Service provides the ability to acquire strategy management data. SAP BusinessObjects Dashboards has its own WSDL parser to consume the Web services WSDLs for strategy management.

For information about the *CubeService* Web service, see [Providing Application Server Data for Other Systems](#) [Page 48]. For information about the *SMDDataService*, see [Providing Strategy Management Data for Other Systems](#) [Page 44].

Prerequisites

You are running SAP BusinessObjects Dashboards.

Process

1. Add users of the *SMDDataService* Web Service to SAP NetWeaver UME.

For information, see [Adding SMDDataService Web Service Users to UME](#) [Page 45].

2. Add users of the *CubeService* Web Service to SAP NetWeaver UME.
For information, see [Adding CubeService Web Service Users to UME](#) [Page 48]
3. Set *Basic Authentication* for the *SMDataService* Web Service.
For information, see [Setting Basic Authentication for the SMDataService Web Service Provider](#) [Page 45].
4. Set *Basic Authentication* for the *CubeService* Web Service.
For information, see [Setting Basic Authentication for the CubeService Web Service Provider](#) [Page 49]
5. Export the WSDL files from SAP NetWeaver.
For information, see [Exporting the WSDL files from SAP NetWeaver](#) [Page 53].
6. Import the strategy management WSDLs into SAP BusinessObjects Dashboards.
For information, see [Configuring Web Service Connections in SAP BusinessObjects Dashboards](#) [Page 54]
7. Map input and output values to the SAP BusinessObjects Dashboard Excel spreadsheet.
For information, see [Mapping Input and Output Values to the SAP BusinessObjects Dashboards Excel Spreadsheet](#) [Page 55].
8. Configure SAP BusinessObjects Dashboard Components to use data from the Excel Spreadsheet.
For information, see [Configuring SAP BusinessObjects Dashboards Components to use Data from the Excel Spreadsheet](#) [Page 56]



5.5.1 Exporting the WSDL files from SAP NetWeaver

Prerequisites

You are the administrator of SAP NetWeaver.

You have added users of the *SMDataService* and *CubeService* Web Service to SAP NetWeaver UME. You have set Basic Authentication for the *SMDataService* Web Service and *CubeService* Web Service.

Procedure

1. Start *Web Services Administration* in SAP NetWeaver using the following URL:
`http://<nw_server>:<port>/nwa`
2. Choose **SOA** → *Application and Scenario Communication* → *Single Service Administration*.
3. Select the *Service Definitions* tab if it is not currently selected.
4. In the *Find* text box, enter *iSMDataService* and click *Go*.
5. In the *WSDL Port Type Name* list, select *iSMDataService* if it is not already selected.

6. Click the *WSDLs* tab.
7. Click the WSDL link for *iSMDDataService*. A new browser window appears with *iSMDDataService* WSDL data.
8. If you see a URL like this in the address bar of the browser and it contains the `&model=ws_policy` parameter as shown below, edit the URL to remove the text `&model=ws_policy`:

```
http://<server>:<port>/strategyServer/SMDDataServiceService?wsdl
&model=ws_policy
```

The URL must end in WSDL. The URL should look like this:

```
http://<server>:<port>/strategyServer/SMDDataServiceService?wsdl
```

9. Refresh the browser after you edit the URL. The browser window displays the complete *SMDDataService* WSDL data.
10. Click **File** → **Save As** and save the opened WSDL file using the `.wsdl` extension to a directory accessible by SAP BusinessObjects Dashboards.
11. From the list of services, select *CubeServiceInterface*.
12. Click the *WSDLs* tab.
13. Click the WSDL links for *CubeService*.

A window appears with *CubeService* WSDL data.

14. Click **File** → **Save As** and save the opened WSDL file using the `.wsdl` extension to a directory accessible by SAP BusinessObjects Dashboards.



5.5.2 Configuring Web Service Connections in SAP BusinessObjects Dashboard

You must configure the Web Service connections to import the strategy management WSDLs to be used in SAP BusinessObjects Dashboard.

Prerequisites

You have prepared the WSDL files for importing into SAP BusinessObjects Dashboard.

Procedure

1. Start Data Manager for SAP BusinessObjects Dashboard and go to **Data** → **Connections**.
2. Click *Add*.
3. Click *Web Service Connections* and type a name for the connection.
4. In the *WSDL URL* text box, type the location of the exported *SMDDataService* WSDL file in the local directory:


```
<drive>:\<path>\SMDDataService.wsdl
```
5. Click *Import*.

All the available Web methods and parameters appear in the *Data connection* window.

6. In the *WSDL URL* text box, type the location of the exported *CubeService* WSDL file in the local directory:

```
<drive>:\<path>\CubeService.wsdl
```

7. Click *Import*.

All the available Web methods and parameters appear in the *Data connection* window.



5.5.3 Mapping Input and Output Values to an Excel Spreadsheet

You need to map the input values and output values to an Excel spreadsheet provided by SAP BusinessObjects Dashboard. Then SAP BusinessObjects Dashboard can read from the spreadsheet and display the results.

Prerequisites

You have imported the strategy management WSDL files in SAP BusinessObjects Dashboard.

You are displaying the Data Manager.

Procedure

1. In Data Manager, select an *Input Value* and click the *Read From* icon.
2. In the *Select a Range* box, enter the cell location where the input parameter is set.
3. Select an *Output Value* and click the *Insert In* icon.
4. In the *Select a Range* box, map the return value to a set of rows in the Excel spreadsheet.

Since data is dynamic, make sure that you allocate adequate rows in the spreadsheet. SAP BusinessObjects Dashboard does not have the option to set data selection dynamic.

If you use SAP BusinessObjects Enterprise authentication, you cannot test the Web services in the Web Service Navigator. There is no way to pass the token in WSNavigator which is out of strategy management control.

To use SAP BusinessObjects Enterprise authentication in SAP BusinessObjects Dashboard, make a Web service call `getFPMSession()` first. The returned session and token then need to be included with other information to build up a SOAP Header. This SOAP header then needs to be included in every subsequent method call in the Advanced Tab of the Web Service Connection definition for a given method call.

The format of the request is as follows:

```
<soapenv:Header>
```

```
<FpmSoapEnv:Header>
```

```
<serializedSession>Enter session value here</serializedSession>
```

```
<fpmLogonToken>Enter session value here</fpmLogonToken>
```

```
</FpmSoapHeader>
```

```
</soapenv:Header>
```

For example:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ssm="http://sap.com/cpm/sm/webservices/ssm/">

<soapenv:Header>

<FpmSoapHeader>

<serializedSession>Enter session value here</serializedSession>

<fpmLogonToken>Enter session value here</fpmLogonToken>

</FpmSoapHeader>

</soapenv:Header>

<soapenv:Body>

<ssm:getAllContext>

<dbNameId>1</dbNameId>

</ssm:getAllContext>

</soapenv:Body>

</soapenv:Envelope>
```



5.5.4 Mapping SAP BusinessObjects Dashboard Components to the Excel Spreadsheet Data

To load data that is returned from Web Services, you must create a mapping between each SAP BusinessObjects Dashboard object and the cells in the Excel spreadsheet.

We recommend that you create a scorecard template using SAP BusinessObjects Dashboard components and test the data. Once the visualization is complete, you can remove the test data and map to the spreadsheet data source.

Prerequisites

You have mapped input and output values to the Excel spreadsheet.

Procedure

1. Double-click the object to display the *Property* window.
2. In the *General* tab, click the icon next to the *Labels* field.
3. In the *Select a Range* box, specify a range in the spreadsheet where the labels are loaded.
4. Now you specify where to insert the data. In the *Insertion Type* dropdown list, select *Row*.
5. Click the icon next to the *Source Data* field. In the *Select a Range* box, specify the range where all the data is available.

6. Click the icon next to the *Destination* field and specify the destination row.

For information about using SAP BusinessObjects Dashboard, see the SAP Help Portal for SAP BusinessObjects Dashboard at <http://help.sap.com> → *Analytics* → *All Products* and search on the product name.



5.6 Providing Strategy Management Data for Webl

You can allow users of Webl to use strategy management application data for ad hoc analysis in Webl. You make the strategy management data available using the Strategy Management ODBO Provider.

The strategy management application allows you to communicate the strategy, communicate targets, monitor, execute, and improve execution by implementing initiatives. You may need to extend this analysis to provide power users with the ability to perform queries and analysis of strategy management information. By accessing strategy management data in Webl, users of Webl can build reports about the strategy and share them across the organization.

Prerequisites

- SAP BusinessObjects 4.0 is installed on a Windows server. The server can be installed on either the same machine with the strategy management system and SAP NetWeaver or a separate Windows server (recommended). For information, see the SAP BusinessObjects 4.0 Guides on the SAP Help Portal at <http://help.sap.com>.
- The SAP BusinessObjects Webl server is installed and configured on the SAP BusinessObjects server. For information, see the SAP BusinessObjects Administrator's Guide in the *SAP BusinessObjects* section of the SAP Help Portal.
- Webl users and system groups are added to SAP BusinessObjects Central Management Console. You might want to create a separate group in SAP BusinessObjects and add all Webl users and system groups who work with strategy management.
- SAP BusinessObjects is set up as the user management system for the strategy management application. This includes the following:
 - The `StrategyGroup` Java System Property for the strategy management application in SAP NetWeaver has the value `strategy`.
- SAP BusinessObjects users are able to access strategy management using SSO via SAML2.

For information, see the *Installation Guide for SAP Strategy Management* on the SAP Help Portal at <http://help.sap.com/bosm101>. Then choose *Configuring the Application and Setting Up the User Management System*.

Process

1. Add users and system groups to the `strategy` group to identify them as strategy management users.

For information, see *Adding Strategy Management Users* in the *Installation Guide for SAP Strategy Management* on the SAP Help Portal at <http://help.sap.com/bosm101>.

2. In SAP NetWeaver, assign the new strategy management users to a role. In the administration application, assign the role to a context.
3. Install and register `SSMProvider.dll` and `SSMProviderEr.dll`.

For information, see [Installing the Strategy Management ODBO Provider](#) [Page 65].

4. Create the universe.

For information, see [Creating the Universe](#) [Page 59].

5. Assign WebI users and groups to the WebI connection.

For information, see [Assigning WebI Users and Groups to the WebI Connection](#) [External].



This step is necessary only if you want to authenticate using Single Sign On (SSO).

6. Export the universe.

For information, see [Exporting the Universe](#) [Page 60].

7. Create a WebI document and generate a report.

For information, see [Creating a WebI Document and Generating a Report](#) [Page 61].



5.6.1 Installing the Strategy Management ODBO Provider

During the strategy management installation, the `\Program files (x86)\SAP Strategy Management\InternetPub\ODBOProvider` directory is created. It contains the two DLLs needed to run the Strategy Management ODBO Provider, and two `bat` files to register and unregister the DLLs.



You need to install the Strategy Management ODBO Provider if you want to display strategy management data in SAP BusinessObjects Analysis, WebI, or SAP Crystal Reports. If you use more than one of these systems, you only need to install and register the DLLs once.

Prerequisites

SAP BusinessObjects is installed on a Windows server.

Procedure

1. In the SAP BusinessObjects Central Configuration Manager, stop the multidimensional analysis service (MDAS).



If you have a SAP BusinessObjects Analysis implementation or a WebI implementation, you must stop the service.

If you have a SAP Crystal Reports implementation, you only need to stop the service if you are using SAP BusinessObjects to access data via the Universe. If you are not accessing data via the OLAP Universe, you do not need SAP BusinessObjects and therefore do not need to stop services.

2. Copy the entire `\ODBOProvider` directory from the Windows server where strategy management is installed to any directory on the SAP BusinessObjects Windows server.
3. Run the `SSMProviderReg.bat` file to register the DLLs.
4. If you are running Strategy Management 10.0 SP5 or earlier, perform the following step; if you are running Strategy Management 10.0 SP6 or later, ignore this step.

Open the registry, and add the following `servletUri` registry setting:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\SAP\SSM\ODBOProvider]
"servletUri"="/strategyServer/ODBOProviderServlet"
```

Note: If the SAP BusinessObjects system is on a 64-bit Windows system, put this setting under `HKLM\SOFTWARE\Wow6432Node\SAP\SSM\ODBOProvider`.

If you already have the following registry setting from a previous 7.5 version of Strategy Management:

```
"servletUri"="/strategy/ODBOProviderServlet"
```

Change it to the following (and if running a 64-bit Windows system for SAP BusinessObjects, put it in

```
HKLM\SOFTWARE\Wow6432Node\SAP\SSM\ODBOProvider:
```

```
"servletUri"="/strategyServer/ODBOProviderServlet"
```

5. In the SAP BusinessObjects Central Configuration Manager, restart the multidimensional analysis service (MDAS).



If you have a SAP BusinessObjects implementation or a WebI implementation, you must start the service.

If you have a SAP Crystal Reports implementation, you only need to start the service if you are using SAP BusinessObjects to access data via the Universe. If you are not accessing data via the OLAP Universe, you do not need SAP BusinessObjects and therefore do not need to start services.



If you ever need to unregister the files, execute the batch job to do the unregistration. If you ever need to upgrade the DLLs, you would have to unregister the existing DLLs, copy over the new ones, and then register the DLLs.



5.6.2 Creating the Universe

A universe is a mapping that lets you determine how you want to see your data. A universe allows you to map labels to a data source.

Prerequisites

You are the administrator of SAP BusinessObjects.

Procedure

1. Start BusinessObjects, open the Universe Design Tool, and log in as the administrator.

2. Click the *Create Universe* button on the toolbar to launch the wizard for creating a universe. If this is your first time running Designer, the wizard appears automatically.
3. In the *Welcome Screen*, click *Begin*.
4. In the *Enter the universe name* box, specify the name for the universe.
5. Click *New* to create a new database connection to the strategy management Provider.
6. In the first *Define a new connection* window, click *Next*.
7. In the *Define a new connection* window for database middleware selection, do the following:
 1. In the hierarchical list, select *Strategy Management*.



Strategy Management appears because you defined this value in the `oledb_olap.sdo` file.

2. In the *Connection Type* dropdown list, specify whether to make this a secured connection.
 3. In the *Connection Name* box, specify a connection a name.
 4. Then click *Next*.
8. In the *Define a new connection* window for logon parameters, do the following:
 1. In the *Authentication Mode* dropdown list, select the method of authentication such as single sign or user specified username and password.
 2. Specify the username, password, and server for the connection, and then click *Next*.
9. In the *Define a new connection* window for catalog/database parameters, display the items in the *SSMCatalog* folder, select the performance management context/model associated with the connection, and then click *Next*.



You can only select performance management models in the *SSMCatalog* folder. Performance management models contain measures that are used as the basis of KPIs.

If you want to view model information such as dimensions, attributes, and measures in the Application Server model, select the model in the *PASCatalog* folder.

10. In the *Define a new connection* window for configuration parameters, use the default settings and click *Finish*.
11. In the *Define the Universe Parameters* wizard, click *Next*.
12. In the final wizard screen, click *Finish*.
13. Click *Save*.



5.6.3 Exporting the Universe

You must export the Universe before you can create a WebI document and generate a report

Prerequisites

You have created the Universe.

Procedure

1. In Designer, choose  *File* → *Export* .
2. Choose the appropriate domain and click *OK* to finish the export.



5.6.4 Creating a WebI Document and Generating a Report

After you export the universe, you can create a WebI document and generate a report.

Prerequisites

You have exported the universe.

Procedure

1. Start BusinessObjects InfoView and log in with appropriate user name/password.
2. Create a Web Intelligence Document by choosing  *Document List* → *New* → *WebIntelligence Document* .
3. Associate the document with the universe you just created by selecting the universe. The WebI main window appears.
4. From the *Data* tab, drag one or more items from the hierarchical list to the *Result Objects* and *Query Filters* panels as appropriate.
5. Click *Run Query* to generate the report.
6. Use the *Data*, *Template*, and *Map* tabs and toolbar buttons to edit the report as appropriate.

5.7 Providing Strategy Management Data for Crystal Reports

You can allow users of SAP Crystal Reports to use strategy management application data in Crystal Reports. You make the strategy management data available using the Strategy Management ODBO Provider or using Query As A Web Service (QaaWS).

Prerequisites

SAP BusinessObjects 4.0 is installed on a Windows server if you want to access data via the OLAP Universe. The server can be installed on the same machine with the strategy management system and SAP NetWeaver or a separate machine (recommended). For information, see the SAP BusinessObjects Guides on the SAP Help Portal at <http://help.sap.com>.

If you do not want to access data via the OLAP Universe, you can omit the SAP BusinessObjects installation.

SAP Crystal Reports is installed and configured. The Report Designer/Report Engine is installed on the server. For information, see the SAP Crystal Reports Installation Guide on the SAP Help Portal at <http://help.sap.com>.

Features

You can use strategy management data in SAP Crystal Reports using either of these methods:

- Use the Strategy Management ODBO Provider.

For information, see [Providing Data Using the Strategy Management ODBO Provider](#) [Page 65].

- Use Query As A Web Service (QaaWS).

For information, see [Providing Data Using Query As A Web Service](#) [Page 62].



5.7.1 Providing Data Using Query As A Web Service

Process

1. Create a QaaWS from a Universe.
2. Create a New Query.
3. Create a Report Using QaaWS for Crystal Reports.



5.7.1.1 Creating a QaaWS from a Universe

The first step to provide strategy management information in Crystal Reports is to create a QaaWS from a Universe.

Prerequisites

You have created a Universe.

For information, see [Creating the Universe](#) [Page 59].

Procedure

1. Open the *Query As A Web Service* (QaaWS) application.
2. If this is the first time you are using Qaaws, create a new host by clicking the *Host* button in the *Select your credentials* dialog box.
 1. In the *Manage Hosts* dialog box, click *Add*.
 2. Specify a name for the host, the URL where it is located, IP address where CMS is located, and the username.
 3. From the *Authentication* dropdown list, select the appropriate authentication type, and then click *OK*.
3. In the *Select your credentials* dialog box. specify the system, username, and password to log in.



5.7.1.2 Creating a Query

Once you create a query, you can create a report using QaaWS for Crystal Reports.

Prerequisites

You have created a QaaWS from the Universe.

Procedure

1. In the *Query as a Web Service* window, choose **Query** → **New** → **Query**.
2. Enter the *Web Service Name* and click *Next*.
3. Select a Universe on which to build the query and click *Next*.
4. Build the query by dragging objects from the hierarchical list to the *Result Objects* and *Filter Objects* section and click *Next*.

You see a preview of the query results.

5. Click *Publish*.

The Query is in the *Query as a Web Service* window and there is a URL for it.



5.7.1.3 Creating a Report Using QaaWS for Crystal Reports

Prerequisites

You have created a query using Query As A Web Service.

You are running Crystal Reports.

Procedure

1. In Crystal Reports, choose **File** → **New** → **Standard Report**.
2. In the *Standard Report Creation Wizard* window, choose *XML and Web Services* and then select *Make New Connection*.
3. In the *XML and Web Services* main window, select *Use Web Service Data Source* and click *Next*.
4. In the next window, select *Use HTTP(s) WSDL*.
5. In the *HTTP(S) WSDL URL* box, enter the WSDL URL you created in QaaWS, and then click *Next*.
6. In the next window, provide a valid user ID and password or leave it blank if the user ID and password are stored in the Universe. Then click *Next*.
7. In the next window, from the *Service* dropdown list, select the service name. This is the same as the Web Service name you specified when creating a query.
8. In the *Port* dropdown list, select *QueryAsAServiceSoap*.
9. In the *Method* dropdown list, select *runQueryAsAService*. Then click *Finish*.

You see the new connection.

10. Select the table *runQueryAsAServiceResponse/table/row* and click the right arrow (>) button to add it to the selected table panel. Then click *Next*.
11. In the *Enter Values* dialog box, enter the administrator logon for logging into SAP BusinessObjects Enterprise (and the Universe) in the text box for *parameters.runQueryAsAService.login*.
12. Enter the password in the text box for *parameters.runQueryAsAService.password*.
13. Click *OK*.
14. (Optional) Group the information on the report, or click *Next* or *Finish*.
15. (Optional) Include a chart in the report, or click *Next* or *Finish*.
16. (Optional) Select a subset of information to display, or click *Next* or *Finish*.
17. (Optional) Select a template for the report and click *Finish*.



5.7.2 Providing Data Using the Strategy Management ODBO Provider

The strategy management ODBO Provider gives you access to the following:

- The AS Adapter allows access to the measures (based on attributes and dimensions) in the Application Server model.

You can also see hierarchies, dimension members, and other aspects of dimensions.

- The SM Adapter allows access to the strategy dimension, which represents the strategy management dimensions *Scorecard* and *Initiative*. Scorecard detail not relating to the KPI such as comments are not presented.

The *Scorecard* dimension presents the KPI measure values for Actual, Target, Score, Status, Trend, Trend Status, and Gap for perspectives, objectives, and KPIs. The *Initiative* dimension presents the KPI status measure for initiatives. Other initiative details such as actual, budget, target budget, and comments are not presented.

To retrieve additional information about scorecards, initiatives, and comments, you must use the strategy management Web services.

Process

1. Install and register `SSMProvider.dll` and `SSMProviderEr.dll`.

For information, see [Installing the Strategy Management ODBO Provider](#) [Page 65].

2. Create an OLAP connection in Crystal Reports for strategy management.

For information, see [Creating an OLAP Connection and Setting up a Report](#) [Page 66].



5.7.2.1 Installing the Strategy Management ODBO Provider

During the strategy management installation, the `\Program files (x86)\SAP Strategy Management\InternetPub\ODBOProvider` directory is created. It contains the two DLLs needed to run the Strategy Management ODBO Provider, and two `bat` files to register and unregister the DLLs.



You need to install the Strategy Management ODBO Provider if you want to display strategy management data in SAP BusinessObjects Analysis, WebI, or SAP Crystal Reports. If you use more than one of these systems, you only need to install and register the DLLs once.

Prerequisites

SAP BusinessObjects is installed on a Windows server.

Procedure

1. In the SAP BusinessObjects Central Configuration Manager, stop the multidimensional analysis service (MDAS).



If you have a SAP BusinessObjects implementation or a WebI implementation, you must stop the service.

If you have a SAP Crystal Reports implementation, you only need to stop the service if you are using SAP BusinessObjects to access data via the Universe. If you are not accessing data via the OLAP Universe, you do not need SAP BusinessObjects and therefore do not need to stop services.

2. Copy the entire `\ODBOProvider` directory from the Windows server where strategy management is installed to any directory on the SAP BusinessObjects Windows server.
3. Run the `SSMProviderReg.bat` file to register the DLLs.
4. Open the registry, and add the following `servletUri` registry setting:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\SAP\SSM\ODBOProvider]
"servletUri"="/strategyServer/ODBOProviderServlet"
```

Note: If the SAP BusinessObjects system is on a 64-bit Windows system, put this setting under `HKLM\SOFTWARE\Wow6432Node\SAP\SSM\ODBOProvider`.

If you already have the following registry setting from a previous 7.5 version of strategy management:

```
"servletUri"="/strategy/ODBOProviderServlet"
```

Change it to the following (and if running a 64-bit Windows system for SAP BusinessObjects, put it in

```
HKLM\SOFTWARE\Wow6432Node\SAP\SSM\ODBOProvider:
```

```
"servletUri"="/strategyServer/ODBOProviderServlet"
```

5. In the SAP BusinessObjects Central Configuration Manager, restart the multidimensional analysis service (MDAS).



If you have a SAP BusinessObjects Analysis implementation or a WebI implementation, you must start the service.

If you have a SAP Crystal Reports implementation, you only need to start the service if you are using SAP BusinessObjects to access data via the Universe. If you are not accessing data via the OLAP Universe, you do not need SAP BusinessObjects and therefore do not need to start services.



If you ever need to unregister the files, execute the batch job to do the unregistration. If you ever need to upgrade the DLLs, you would have to unregister the existing DLLs, copy over the new ones, and then register the DLLs.

5.7.2.2 Creating an OLAP Connection and Setting up a Report

You must create an OLAP connection in Crystal Reports for strategy management.

Prerequisites

You have installed the strategy management ODBO Provider.

Procedure

1. Start Crystal Reports.
2. Choose **File** → **New** → **OLAP Cube Report**.
3. In the *OLAP Data* dialog box, click *Select Cube*.
4. In the *OLAP Connection Browser* dialog box, click *Add*.
5. In the *Connection Properties* dialog box, make sure that *Strategy Management* already appears in the *Server Type* field.
6. In the *Caption* text box, enter a name for the connection.
7. Click the *OLAP Server* option and specify the OLAP Server's *Server Name:port*, *User Name*, and *Password*, and then click *OK*.
8. In the *OLAP Connection Browser* dialog box, select the context from the *SM* catalog and click *Open*.



You can only select performance management models in the *SM* catalog. Performance management models contain measures that are used as the basis of KPIs. If you want to view model information such as dimensions, attributes, and measures in the Application Server model, select the model in the *AS* catalog.

The *OLAP Data* window appears with the name of the selected model, the provider name, and the server IP and port.

9. Click *Next* to specify the rows and columns.
10. In the *Rows/Columns* dialog box, add the dimensions to the appropriate *Rows* and *Columns* lists. Use the *Select Row Members* button and *Select Column Members* button to display specific members of the dimensions.
11. Click *Next* to specify the slice and page for the report.
12. In the *Slice/Page* dialog box, specify what to do with the dimensions not displayed in the report. In the *Slice* list, select dimensions, and then click *Select Slice* to specify the member of the dimension to use to slice (filter) the report.

In strategy management application, these are called filter dimensions. In Crystal Reports, these are called Slices.

13. Click *Next* to specify the style of the report.
14. In the *Style* dialog box, select a predefined style and click *Next*.

15. In the *Chart* dialog box, specify the type of chart to display and click *Finish*. The report is displayed.



5.8 Using an Application Server Fiscal Calendar for Queries in SAP BusinessObjects Enterprise

You can display the standard calendar for queries in SAP BusinessObjects Enterprise tools or you can use the fiscal calendar that users have specified in Application Server to use for models with fiscal settings. By default, the standard calendar is used.

Prerequisites

You are the administrator of SAP NetWeaver.

Procedure

1. Start the SAP NetWeaver Administrator.
2. Log on as administrator with the global password you provided when you installed NetWeaver.
3. Click *Configuration Management*.
4. Click the *Infrastructure* tab.
5. Select *Java System Properties*.
6. In the *Templates* section, select the template that corresponds to your strategy management installation.
7. Click the *Applications* tab.
8. In the *Name* column, type *strategy* and press **Enter** to list the strategy applications.
9. Select the name *xapps~cpm~sm~strategymanagement*.
10. In the *Extended Details* section, click inside the *Name* text box and enter *odbo.timeprovider*.
11. Specify the value *PAS*. The date ranges and calculations are done using the fiscal calendar. The default setting is *Calendar*.
12. Restart SAP NetWeaver to clear the previously cached data.



6 Application Server Configuration Files

The key configuration files of Application Server include:

- LSSERVER.INI
- LSDAL.INI
- LSDAL.CNF



6.1 LSSERVER.INI Settings

The LSSERVER.INI file reflects only those options that you installed on your system, so all of the following entries might not appear in your LSSERVER.INI file.



6.1.1 [Microsoft Windows] settings

The [Windows] section contains options related to general path and other miscellaneous setups.

Features

VERSION=version

The version of Application Server that is currently installed.

SERVER=servername

The name of the UNIX or Microsoft Windows server. If no name is specified, Application Server runs either the client standalone version or the client/server version, depending upon whether a server name is in the logon window.

DBPATH=<DRIVE>:\users\default\Home;<DRIVE>:\users\default\Data

Defines the path Application Server searches for models that are not set up as environmental variables. This option is used for standalone and LAN installations and Microsoft Windows clients running client/server.

DBHOME=<DRIVE>:\users\default\Home

Defines the Application Server home directory, which is the location of external files (for example, output buffer files, traces, database dumps, and so on) if there is no separate environment variable specifically defining the files. Each user must have their own directory. Although this directory can be on a network, it cannot be a shared directory.

PAGEDB=<DRIVE>:\users\default\Home\PAGEDB

Defines the location and name of the Application Server output window page file.

PAGEDBSIZE=25000

Defines the maximum size of the Application Server output window page file, PAGEDB. The default value is 25,000.

CMDSEP=|

Indicates the character to use as the Application Server command separator function. The default is the bar (|) symbol. Many other characters conflict with the normal function of

Application Server. However, the bar (|) symbol, the exclamation point (!), and the tilde (~) do not clash with standard characters in filenames and the Application Server command language.

LOCKFILE=<DRIVE>:\users\default\Data\LOCKFILE

Defines the location of the Application Server lockfile. This option is used for standalone and LAN installations and Microsoft Windows clients running client/server. The lockfile must be in the same location as MASTERDB. Therefore, if MASTERDB resides on a network drive, the lockfile must reside on the same network drive.



All users using the same MASTERDB must use the same lockfile, otherwise serious database corruption could occur.

WINDOWPOS=0 0 882 1152

Defines the coordinates of the *Application Server* window.

EDITFULLSCREEN=0

Indicates the maximum state of the *Editor* window.

1 = *Editor* window is maximized.

0 = *Editor* window is not maximized (default).

EDITWINXPOS=256

Defines the location of the *Edit* window X Coordinate.

EDITWINYPOS=192

Defines the location of the *Edit* window Y Coordinate.

EDITWINWIDTH=896

Defines the width location of the *Edit* window.

EDITWINHEIGHT=576

Defines the height location of the *Edit* window.

NCSTRACE=1

Add this line to enable end-to-end tracing and heartbeat availability monitoring. This is useful when installing and configuring Solution Manager.

You must restart the JPIP session to allow the changes to take effect. To do so, access the JPIP session monitor using the *Tools* utility at

`http://<nw_server>:<port>/strategy/tools.`

To perform end-to-end tracing, you need to use the SAP Client plug-in for Internet Explorer and start a transaction using the normal procedures for end-to-end tracing.

NCSLOG=INFO | WARNING | ERROR | FATAL

Add this line to enable logging for diagnostic purposes. Specify the appropriate severity level by using a value of either INFO, WARNING, ERROR, or FATAL.

If you specify *INFO*, you get the most information (all *INFO*, *WARNING*, *ERROR*, and *FATAL* status messages). If you set it to *WARNING*, *ERROR*, or *FATAL*, you only get messages of that type.

You must restart the JPIP session to allow the changes to take effect. To do so, access the JPIP Session Monitor using the *Tools* utility at
`http://<nw_server>:<port>/strategy/tools.`

All of the generated files are *<uuid>.GLF* files, which you can view using a text editor or SAP's log viewing tool. The files exist in `\Logs` directory of the Application Server installation directory.

Since model connections generate their own log or trace file, this setting is useful for pre-production, but not for production systems.

To turn off logging, you can comment out the line.

DEFAULTMEMORY=

Defines the amount of memory to use in each Application Server session. The default value is 12,000 kilobytes (Kb) on Microsoft Windows servers.



This setting can be overridden at the command line, or from within a procedure. For more information about `DEFAULTMEMORY`, see the `SET MEMORY` command in Application Server online Help.

MASTERDB=<DRIVE>:\users\default\Data\MASTERDB

Defines the location of the required model, `MASTERDB`. This option is used for standalone installations only.

TBDB=<drive>:\users\default\Data\TBDB.ENG

Defines the location of the required model, `TBDB`. This option is used for standalone and LAN installations only.

INITIAL=C:\users\default\Data\INITIAL

Defines the location of the distribution model, `INITIAL`. This option is used for standalone and LAN installations only.

APLIB=C:\users\default\Data\APLIB

Defines the location of the distribution model, `APLIB`. This option is used for standalone and LAN installations only.

SMREPORT=C:\users\default\Data\SMREPORT

Defines the location of the distribution model, `SMREPORT`. This option is used for standalone and LAN installations only.

JUICE=C:\users\default\Data\JUICE.ENG

Defines the location of the distribution model, `JUICE`. This option is used for standalone and LAN installations only.

TSPPROMPT=Yes

Controls the display of the logon dialog box.

Yes = Displays a logon dialog box, even if the username and password have been specified (default).

No = The logon screen does not appear unless required.

EXITCLEAR=Yes

Specifies how Application Server should be closed when exiting from an application.

Yes = Clears the `work` database (default).

No = Does not clear the `work` database.

SHAREDMEMORY=NO

Indicates whether shared memory is used.

Yes = Shared memory is used.

No = Shared memory is not used. Required for Microsoft Windows server installations (default).



6.1.2 [Server] Settings

The `[Server]` section contains options related to the Application Server when implemented on a Microsoft Windows server or UNIX Server.

Features

SERVERPATH=C:\Program Files (x86)\SAP Strategy Management\ApplicationServer

Defines the location (directory) of the Application Server components for client/server operation.



6.1.3 [XXXXX] Settings

The `[XXXXX]` section contains options related to the Application Server when it is connecting to a remote UNIX or Microsoft Windows server machine. When you create a remote server using Application Server, a new `[XXXXX]` section is created, where `[XXXXX]` is the name of the remote server.

Features

[XXXXX]

Indicates the server name.

PROTOCOL=TCP

Indicates the selected communications protocol (must be TCP).

SERVICE=PILOT

Defines the `lssmap` file service name to connect to.

In UNIX, this parameter is the name of the shell script found in the `lssmap` file that is run when you connect to UNIX. It is case sensitive.

CURSOR=one of LSSCMPTR or LSSCOMMS or NONE

Indicates cursor to use when indicating client/server communication.

USERNAME=xxx

UNIX or Microsoft Windows server user name; not the same as your Application Server logon name.

PASSWORD=password

This value should be empty unless you want to continue to use clear text passwords to the server machine. The line is overlooked if it is empty and you have an `EncryptedPassword` line.

ENCRYPTEDPASSWORD=encrypted_password

Encrypted password to the UNIX or Microsoft Windows server machine running Application Server.

If this line is empty or contains a question mark (?), you are prompted to enter your password every time you start Application Server in a client/server configuration.

When you create a remote server connection in Application Server by choosing *File*, then *New*, then *Remote Server*, the password you enter in the *Create Remote Server* dialog box is stored in encrypted format in this line in `lsserver.ini`. To learn about this, see *Creating a remote server connection* in the Application Server online Help.

LOGFILE=LSSWSOCK.LOG

Specify `LOGFILE` for the output trace file you want to use. The default is `LSSWSOCK.LOG` for `WINSOCK` connections.

LOGLEVEL=BASIC

Specify `LOGLEVEL=BASIC` for function name and return code, or `LOGLEVEL=ADVANCED` for function name, return code, and data.

PORT=8325

Specifies a port setting if other than 8325 (the default).

NTUSERLOGIN=Y

This option is available if you are running Application Server in client/server mode on an NT server.

The default setting of `Y` means that the user is required to logon to the NT security, and have a user ID for that NT domain.

If you specify `NTUSERLOGIN="N"` on the server, the user name and password entries are not validated, and `USERPRIV=N` is also implied for this user. The Application Server process that is started on the server runs in the security context of the NT Local System account.



If you change this setting in `LSSERVER.INI` on the server, you must stop and then restart the SAP SM Listener.

Although the user name and password are not validated on the server, these entries are still required in `LSSERVER.INI` on the client — set both `USERNAME` and `PASSWORD` to `NULL` or `BLANK`.

USERPRIV=Y|N

You can control the user's ability to access network privileges on the server by adding this parameter to this section of the `LSSERVER.INI` file. In NT Server for Intel 4.0, users have network privileges by default.

When `USERPRIV=Y`, user access to network drives is based on the privileges of the user account specified by the User and Password entries in the `LSSERVER.INI` file on the client machine. When `USERPRIV=N`, user access to network drives is based on the privileges of the Local System account on the server machine.

To access files stored on Novell network drives, the universal naming convention (UNC) must be specified in the `DBPATH` variable or in other database variables defined in the `LSSERVER.INI` file.

The ability to connect to NFS mounted drives may depend on the version of the NFS software you are running. Contact Support Services for information.



6.2 LSDAL.INI Settings

The `LSDAL.INI` file contains a dictionary of Link database definitions or Link IDs. The `LSDAL.INI` file reflects only those options that you have installed on your system, so all of the following entries might not appear in your `LSDAL.INI` file. Use Link to modify this file.



6.2.1 [ConnectionID] Settings

This section contains options related to Link IDs that you have created.



6.2.2 [Enable] Settings

The `[Enable]` section contains options you have currently installed.



6.2.3 [Settings] Settings

The `[Settings]` section contains various Link configuration settings.

Features

LastSource=IDname

The name of the last Link ID to which you successfully connected.

Version=v<x>.<x>

The version number of the currently installed product.

[linkid] settings

The Link Configurator uses the information in this section to connect to the selected database. When you create a Link ID in Application Server, the values you define for the connection are stored in this section, where `linkid` is the name you have provided for that Link ID.

ArraySize=n

You can add this parameter to the `[linkid]` section of the `LSDAL.INI` file to specify a default array size for the Link Configurator to use when fetching records from the RDBMS

with a PEEK, READ, or CONSTRUCT command. For example, if you specify `ArraySize=200`, then 200 records are fetched at a time during the READ process.

For `n`, specify a number between 1 and 32,768 for the size of the array you want to fetch from the RDBMS during a PEEK, READ, or CONSTRUCT process.

If you also use the `ARRAYSIZE n` keyword on the PEEK, READ, or CONSTRUCT commands, then the `ArraySize` parameter in `LSDAL.INI` is overridden by the `ARRAYSIZE` command keyword. See the Application Server online Help for information about the `ARRAYSIZE` command keyword.

If you do not add the `ARRAYSIZE` parameter in the `[linkid]` section of the `LSDAL.INI` file, and you also do not specify the `ARRAYSIZE` keyword on the PEEK, READ, or CONSTRUCT commands, then the Link Configurator uses the default value of 100 to fetch 100 records at a time from the RDBMS during a PEEK, READ, or CONSTRUCT.



Some drivers do not support array fetching, such as the Microsoft ODBC driver for MS Access. In those cases, the array size has a value of 1, and the Link Configurator only runs single row fetches.

The best array size may be different on different systems and networks, so you should experiment with array size numbers until you find the optimal value.



6.3 LSDAL.CNF Settings

The `LSDAL.CNF` file contains template Link definitions. It is read when you create a new Link ID in Application Server. Other ODBC-related information is taken from the Registry entries created during installation of Application Server.

