



Your PDF Guides

You can read the recommendations in the user guide, the technical guide or the installation guide for BUSINESS OBJECTS UNIVERSE BUILDER. You'll find the answers to all your questions on the BUSINESS OBJECTS UNIVERSE BUILDER in the user manual (information, specifications, safety advice, size, accessories, etc.). Detailed instructions for use are in the User's Guide.

User manual BUSINESS OBJECTS UNIVERSE BUILDER
User guide BUSINESS OBJECTS UNIVERSE BUILDER
Operating instructions BUSINESS OBJECTS UNIVERSE BUILDER
Instructions for use BUSINESS OBJECTS UNIVERSE BUILDER
Instruction manual BUSINESS OBJECTS UNIVERSE BUILDER

BusinessObjects Universe Builder
Guide

BusinessObjects Universe Builder Guide



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS UNIVERSE BUILDER user guide](http://yourpdfguides.com/dref/3796642)
<http://yourpdfguides.com/dref/3796642>

.....
.....
.....
.....
24 BusinessObjects Universe Builder Guide 3 Contents Exporting a universe to DB2 Cube Views formats.....

.....
.....
.....
.....
.....25 How to identify non-matching universe metadata...

.....
.....
.....
.....
.....26 Mapping specific SQL expressions.....

.....
.....
.....
.....
....28 How universe structures are mapped to DB2 Cube Views metadata.....

..30 Universe to cube model...

.....
.....
.....
.....
....30 Class to dimension.....

.....
.....
.....
.....
...31 Fact table to Fact..

.....
.....
.....
.....
.....31 Measure to measure.....

.....
.....
.....

.....
.....
.....
.....

.....
.....
.....

.32 Dimension and detail object to attribute.....

.....
.....
.....
.....

.....33 Dimension and detail relationship to attribute relationship..

.....
.....

.....33 Default hierarchy to hierarchy...

.....
.....
.....
.....

.....
.....
.....

.....34 Custom hierarchy to hierarchy.

.....
.....
.....
.....

.....
.....
.....
.....

34 Join to join.....

.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

..34 Chapter 6 Building universes from Oracle Analytic Services 37 Overview of how you create a universe from Oracle Analytic Services.....38 How is a universe generated from a OLAP cube?...

.....
.....
.....

...38 Analyzing the relational view..

of Designer? Given that you can use either Designer or Universe Builder to create a universe from metadata source, why use Universe Builder and when do you use it? Universe Builder is an independent product. You can update the metadata plugins used to generate universes when newer versions are available from Business Objects. Designer can not be updated with new or updated metadata exchange plugins. The metadata source support is fixed at the level of the installed version. You use either Designer or Universe Builder based on the following: · · You use Designer to create a universe from a metadata source supported at the time of this release. You use Universe Builder to create a universe from a metadata source plugin that has been updated, or to use a plugin for a newly supported metadata source.

BusinessObjects Universe Builder Guide 7 1 Introduction When do you use Universe Builder instead of Designer? 8 BusinessObjects Universe Builder Guide Universe Builder basics 2 2 Universe Builder basics Starting Universe Builder Starting Universe Builder You start Universe Builder from the task bar by clicking the Universe Builder icon in the group of installed Business Objects products for this release. You are prompted to log into the CMS before Designer starts. Once you start Universe Builder, the Universe Builder pane appears. All the metadata sources supported in this BusinessObjects release are listed as icons in the Bridges column to the left of the Universe Builder pane. Each metadata source also appears as a tab in the central part of the pane.

You can activate a metadata source by clicking either its icon to the left, or clicking its tab. 1. Select Universe Builder from the BusinessObjects group of products in the Start menu. The login box for the CMS appears. 2.

Type the following information. This information is normally provided for you by the BusinessObjects administrator. Login information System User Name Password Authentication Description Name of the CMS server. Your repository user name. Your repository password. Your security level 3. Click the OK button. The Universe Builder pane appears. 10 BusinessObjects Universe Builder Guide Building universes from XML metadata sources 3 3 Building universes from XML metadata sources Overview of how you create a universe from XML metadata Overview of how you create a universe from XML metadata You can create or update a universe from XML files conforming to the following data source standards: · · · · · Common Warehouse Model (CWM Relational 1.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS UNIVERSE BUILDER user guide](http://yourpdfguides.com/dref/3796642)
<http://yourpdfguides.com/dref/3796642>

0 Common Warehouse Model OLAP (CWM OLAP) Oracle Warehouse Builder (Oracle WB) Data Integrator IBM DB2 Data Warehouse Center (IBM DB2 DWC) IBM DB2 Cube Views You can also export a universe to a IBM DB2 Cube View XML format file.

You create a universe by following a metadata source wizard, that allows you to connect to a target XML file. You can then select the cube associated with the XML metadata source. You then start a creation wizard that allows you to select cube tables to include in the universe. You select a connection, set creation options and generate the universe. The universe creation process is the same for all XML metadata sources. The universe creation process for a Oracle Analytical Workspaces datasource is different. Once you have selected your connection, a universe creation panel specific to Oracle Analytic Workspaces appears. You create a view on the database, then create a universe from the view. This is described in the chapter Building universes from Oracle Analytic Services. Once you have created the universe from a metadata source, you can modify any of the universe components as for any other universe.

You save and export the universe to the Central Management System (CMS). Once exported to the CMS, the universe is then available to Web Intelligence users to create queries and reports. 12 BusinessObjects Universe Builder Guide Building universes from XML metadata sources XML metadata to universe component mapping 3 XML metadata to universe component mapping The following table shows the BusinessObjects structures that are mapped from information in the XML file: XML information Database Table or view Table business name Column Column business name Column SQL data type Table/column description Foreign key-Primary key relationship (PKTable.PKColumn=FKTable.FKColumn) Universe structure Universe Class, table Class name Object, column Object name Object type Class/object description Join expression Selecting a XML data source You select a metadata source (an XML file) to create or update a universe from the Universe Builder panel.

1. Start Universe Builder. The Universe Builder panel appears. 2. Click the icon or tab for your metadata source.

For example to create a universe from a CWM relational 1.0 metadata source, you can click either the icon to the left of the Universe Builder panel or the tab at the top of the Universe Builder panel. BusinessObjects Universe Builder Guide 13 3 Building universes from XML metadata sources Creating a universe from a XML data source 3. Do one of the following: · t directs the bridge to the selected metadata through the original XML file. You can browse to and select a folder 16 BusinessObjects Universe Builder Guide Building universes from XML metadata sources Viewing the trace log 3 Viewing the trace log · Example: Setting trace log options The Universe Builder creates a log file that contains trace information for view and universe creation. The log file is generated in the default folder <INSTALL DIR>\Log Files. You can change the default target folder. You can set the following trace options: Tab Option Description Path to the trace folder. This is the folder that holds the log files when universes are generated. You can browse to and select a folder.

Path to the default folder that stores the XML files used to create universes. You can browse to and select a folder. General Traces File locations Default XML Source File Folder BusinessObjects Universe Builder Guide 17 3 Building universes from XML metadata sources Setting trace log options Tab Option Description Path to the default folder that stores the parameter files. These are files created when a universe is created. These files store and reference the selected metadata so that it can be reused for creating or updating other universes. The parameter file does not store the selected metadata. it is a filter that directs the bridge to the selected metadata through the original XML file. Parameter File 1. Select View > Options or click the Options button. The Options page appears.

2. Set options as required. 3. Click OK. 18 BusinessObjects Universe Builder Guide Updating XML metadata source universes 4 4 Updating XML metadata source universes Updating a universe from a XML metadata source Updating a universe from a XML metadata source You select a metadata source that has been updated.

This is the metadata source that has been used to create a universe. The source has been updated, and now you want to update the universe with the same modification. A universe update wizard takes you through the steps needed to update the universe. 1. Start Universe Builder and select a metadata source.

See the section Selecting a XML metadata source for information on doing this. 2. Click the source database icon under the XML metadata source name in the Universe Builder panel 3. Do one of the following: · Select File > Update Universe. · Click the Update Universe button. The Universe update wizard starts. 4. Click Next. The universe file page appears. 5.

Click the Browse button and select a universe. This is the universe that you want to update from the selected XML metadata source. Click Next. The universe elements page appears. The available database tables and columns are listed in the left pane. Tables that have been added or modified are indicated with a red check mark 6. Select one or more tables and columns and click the right arrow to populate the right pane with the tables that have been modified. The tables and columns in the right pane are those that appear in the generated universe. You can use the arrow buttons to add and remove tables from the universe pane as necessary. 7.

Click Next. The elements to update page appears. 8. Click the Advanced button. The Options box appears.

9. Click the Properties tab and select radio buttons to keep or replace Business Name, Mapping Expression, Object Type, and Lineage properties. 20 BusinessObjects Universe Builder Guide Updating XML metadata source universes Updating a universe from a XML metadata source 4 10. Click the Description tab and select radio buttons to keep, concatenate, or replace Business and Technical descriptions. 11.

Click the File Location and General tabs to change parameter file paths, then click Next. The universe generation summary page appears. It lists a summary of the options that you have selected during the wizard. 12. Click Finish. The updated universe appears in the universe and structure panes of Designer.

BusinessObjects Universe Builder Guide 21 4 Updating XML metadata source universes Updating a universe from a XML metadata source 22 BusinessObjects Universe Builder Guide Exporting a universe to a XML file format 5 5 Exporting a universe to a XML file format Overview of exporting universes to XML file formats Overview of exporting universes to XML file formats You can export a universe to IBM DB2 cube View XML format file.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS UNIVERSE BUILDER user guide](http://yourpdfguides.com/dref/3796642)

<http://yourpdfguides.com/dref/3796642>

The universe definition is exported to a XML file that complies with IBM DB2 Cube Views XML format. This file can then be loaded in IBM DB2 Cube Views using the API or OLAP Center tool. IBM DB2 Cube Views reads the metadata from the XML file, and recommends the appropriate Automatic Summary Table (AST) for future query optimization.

Universe pre-requirements for export to DB2 Cube Views The following list describes the universe pre-requisites necessary for a successful universe export to the XML file: Universe element affected Restrictions · · Universe level re- · · strictions · · Each universe is exported to a Cube Model. A universe must match a single snowflake schema with a single fact table. The universe must contain at least one measure. Links between universes are not supported. Contexts are not taken into account, and not exported. Custom hierarchies: the levels of a custom hierarchy must be grouped in the same class. 24 BusinessObjects Universe

Builder Guide Exporting a universe to a XML file format Exporting a universe to DB2 Cube Views formats 5 Universe element affected Restrictions · · @Select function is the only supported @function. All other @functions are not mapped in the export. Conditions in the Where field of an object definition are not exported. Note: Conditions are not supported in DB2 Cube Views objects, as they are not used for optimization.

Multi-parameter aggregation functions are not exported. Each class must contain objects that are built on the same dimension tables. All the objects (attributes in IBMDB2CV) that are referenced by the same dimension in IBMDB2CV, must be grouped in the same class in the universe. Any other measure in other classes are automatically added to the Fact object in IBMDB2CV. Classes and objects · · · Joins If the left or right column of a join does not match an object in the universe, then an attribute for this column is automatically created and added to the dimension (or fact) that contains the column table.

Exporting a universe to DB2 Cube Views formats You export a BusinessObjects universe to a IBM DB2 Cube Views XML file as follows: 1. Start Universe Builder and click the IBMDB2CubeViews icon or tab. 2. Do one of the following: · Select File > Export Universe. · Click The Export Universe button.

The export wizard starts. 3. Click Next. A universe source file page appears. 4. Browse to and select a universe file, then click Next. BusinessObjects Universe Builder Guide 25 5 Exporting a universe to a XML file format How to identify non-matching universe metadata An OLAP information page appears. 5. Do the following: · Enter a name for the fact table, or you can accept the fact table name by default. · Enter a name for the schema.

· Click Next. The metadata is loaded. A page appears showing the structures that will be exported. 6. Click Next. 7. Type a name for the XML file and click Next. A summary page appears. 8. Verify that the export information is correct, then click Finish.

The XML file is created in the universes folder of your user profile, for example, C:\Documents and Settings\<username>\Application Data\Business Objects\Business Objects <version>\Universes. How to identify non-matching universe metadata This section describes how objects in a universe that do not have a multi-dimensional match in IBM DB2 Cube Views are identified and processed during the export of a universe definition to an XML file. A universe is based on relational metadata that does not have multi-dimensional design constraints. All the objects in a universe do not necessarily match equivalent IBM DB2 Cube Views objects, and do not respect IBM DB2 Cube Views multi-dimensional rules. To correctly match relational structures, Universe Builder must run certain automatic detection processes to identify and define the required and appropriate metadata for IBM DB2 Cube Views.

The affected multi-dimensional objects are described below. 26 BusinessObjects Universe Builder Guide Exporting a universe to a XML file format How to identify non-matching universe metadata 5 Non-matching universe metadata- Description ta Fact The IBM DB2 Cube Views Fact object is automatically built from the set of measures found in the universe. Tables that are not identified as Facts are assumed to be dimension tables. A IBM DB2 Cube Views dimension object is deduced directly from a BusinessObjects class. Dimensions All the objects within a class determine the attributes of the IBM DB2 Cube Views dimension.

The tables inferred by BusinessObjects objects within the class, are detected by parsing the object's Select field. Attributes are not directly deduced from the columns of tables in the universe. Candidate attributes are detected and identified from the following information: · BusinessObjects objects within a class. · BusinessObjects objects referenced in Select field of other BusinessObjects objects by the @Select statement. @@@@All classes are searched for measure objects. @@@@Name and Business name Concat character is "_" Left column Right column Complex expression: For each simple expression the left and right columns are identified. Each simple expression maps to an attribute pair. Left attribute Right attribute BusinessObjects Universe Builder Guide

35 5 Exporting a universe to a XML file format How universe structures are mapped to DB2 Cube Views metadata 36 BusinessObjects Universe Builder Guide Building universes from Oracle Analytic Services 6 6 Building universes from Oracle Analytic Services Overview of how you create a universe from Oracle Analytic Services Overview of how you create a universe from Oracle Analytic Services You create a Oracle OLAP universe by clicking the Oracle OLAP icon or tab and selecting File > Metadata sources. The Oracle OLAP Universe Builder wizard starts.

You select a connection. The cubes available to the connection appear. You select a cube, and set view and universe creation options. Related Topics · · How is a universe generated from a OLAP cube? on page 38 How to create a Oracle OLAP universe on page 48 How is a universe generated from a OLAP cube?

When you create a universe with the Oracle OLAP Universe Builder, it is automatically set up for SQL access to Oracle Analytic Workspaces. BusinessObjects Oracle OLAP Universe Builder performs the following main tasks: · · · · Inserts the relational fact view in the universe as a real view or as Derived Table. Add aliases to represent the dimension levels and hierarchies Joins the relational view to the dimension tables with regular joins and shortcut joins. The expressions of the joins are specific to this solution. Creates a class of objects for each Cube dimension and an object for each level of the dimension.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS UNIVERSE BUILDER user guide](http://yourpdfguides.com/dref/3796642)
<http://yourpdfguides.com/dref/3796642>

Creates a subclass for each hierarchy if a dimension has more than one hierarchy. Multi-hierarchy dimensions are supported in the view definition and in the universe.

Define aggregate navigation to resolve object incompatibility that results from the multi-hierarchy dimensions. Defines object expressions using the AggregateAware function to handle the Aggregation Navigation · · 38 BusinessObjects Universe Builder Guide Building universes from Oracle Analytic Services Analyzing the relational view 6 · · Transforms objects that map real dimension members (Identifiers) to Details of objects that represent the member descriptions. Creates measure objects Analyzing the relational view BusinessObjects Oracle OLAP Universe Builder generates views that call the OLAP_TABLE function to map the view columns to the hierarchies of the dimensions and measures of the Cube. The generated script has the following form:

```
CREATE VIEW BOBJ_FK_UNITS_CUBE_VIEW AS SELECT * FROM TABLE(OLAP_TABLE('GLOBAL_AW2.TEST DURATION session',',','&LIMIT_MAP' LIMIT_MAP is a variable that stores the text of the limit_map parameter of OLAP_TABLE.
```

This text is generated by Oracle OLAP Universe Builder. Here is an example of limit_map parameter: DIMENSION GLOBAL_AW2.TEST!FK_TIME WITH HIERARCHY GLOBAL_AW2.TEST!FK_TIME_PARENTREL (FK_TIME_HIERLIST('CALENDAR')) LEVELREL

```
FK_TIME_YEAR,FK_TIME_QUARTER,FK_TIME_MONTH FROM GLOBAL_AW2.TEST!FK_TIME_FAMILYREL USING GLOBAL_AW2.TEST!FK_TIME_LEVELLIST LEVELREL FK_TIME_YEAR_DESC,FK_TIME_QUARTER_DESC,FK_TIME_MONTH_DESC FROM GLOBAL_AW2.TEST!FK_TIME_FAMILYREL USING GLOBAL_AW2.TEST!FK_TIME_LEVELLIST LABEL GLOBAL_AW2.TEST!FK_TIME_LONG_DESCRIPTION ATTRIBUTE FK_TIME_LEVEL FROM GLOBAL_AW2.TEST!FK_TIME_LEVELREL DIMENSION GLOBAL_AW2.TEST!FK_CUSTOMER WITH HIERARCHY GLOBAL_AW2.TEST!FK_CUSTOMER_PARENTREL (FK_CUSTOMER_HIERLIST('MARKET_SEGMENT')) INHIERARCHY GLOBAL_AW2.TEST!FK_CUSTOMER_INHIER LEVELREL null,null,null,FK_CUSTOMER_TOTAL_MARKET,FK_CUSTOMER_MARKET_SEGMENT, FK_CUSTOMER_ACCOUNT,FK_CUSTOMER_SHIP_TO FROM GLOBAL_AW2.TEST!FK_CUSTOMER_FAMILYREL USING GLOBAL_AW2.TEST!FK_CUSTOMER_LEVELLIST LEVELREL null,null,null,FK_CUSTOMER_TOTAL_MARKET_DESC,FK_CUSTOMER_MARKET_SEGMENT_D01, FK_CUSTOMER_ACCOUNT_DESC,FK_CUSTOMER_SHIP_TO_DESC FROM GLOBAL_AW2.
```

```
TEST!FK_CUSTOMER_FAMILYREL USING GLOBAL_AW2.TEST!FK_CUSTOMER_LEVELLIST LABEL GLOBAL_AW2.TEST!FK_CUSTOMER_LONG_DESCRIPTION HIERARCHY GLOBAL_AW2.TEST!FK_CUSTOMER_PARENTREL (FK_CUSTOMER_HIERLIST('SHIPMENTS')) INHIERARCHY GLOBAL_AW2.TEST!FK_CUSTOMER_INHIER BusinessObjects Universe Builder Guide 39 6 Building universes from Oracle Analytic Services What are the shortcut joins in the universe used for? LEVELREL null,null,null,FK_CUSTOMER_ALL_CUSTOMERS, FK_CUSTOMER_REGION,FK_CUSTOMER_WAREHOUSE,null FROM GLOBAL_AW2.TEST!FK_CUSTOMER_FAMILYREL USING GLOBAL_AW2.TEST!FK_CUSTOMER_LEVELLIST LEVELREL null,null,null,FK_CUSTOMER_ALL_CUSTOMERS_DESC, FK_CUSTOMER_REGION_DESC,FK_CUSTOMER_WAREHOUSE_DESC,null FROM GLOBAL_AW2.TEST!FK_CUSTOMER_FAMILYREL USING GLOBAL_AW2.TEST!FK_CUSTOMER_LEVELLIST LABEL GLOBAL_AW2.TEST!FK_CUSTOMER_LONG_DESCRIPTION ATTRIBUTE FK_CUSTOMER_LEVEL FROM GLOBAL_AW2.
```

TEST!FK_CUSTOMER_LEVELREL MEASURE FK_UNITS_CUBE_UNITS AS NUMBER FROM GLOBAL_AW2.TEST!FK_UNITS_CUBE_UNITS ROW2CELL OLAP_CALC What are the shortcut joins in the universe used for? Shortcut joins ensure that BusinessObjects generates SQL for each object combination rather than for each object. BusinessObjects uses shortcut joins when it can omit tables from a query and take a 'shortcut' between two tables that are not directly linked in a hierarchy. For example, based on the following schema: 40 BusinessObjects Universe Builder Guide Building universes from Oracle Analytic Services What are the shortcut joins in the universe used for? 6 if a shortcut join is defined between the QUARTER and OLAPCUBE tables, BusinessObjects does not need to join through the MONTH table to retrieve revenue by quarter. Each table in the time hierarchy (except the lowest-level table) must be joined to OLAPCUBE.

time_level by a shortcut join, as shown below: The join expression must include the expression that will restrict the rows returned from OLAPCUBE; in the case of QUARTER, this is OLAPCUBE.time_level = 'QTR'. To ensure that Designer allows the join, the expression must also reference the MONTH table, which should appear inside comments (because it plays no part in the actual join expression that you are interested in generating). The full join expression is therefore: /* QUARTER.DUMMY */ OLAPCUBE.

time_level = 'QTR' The full list of shortcut join expressions for the example time hierarchy is as follows: Joined tables Expression MONTH, OLAPCUBE /* MONTH.DUMMY */ OLAPCUBE.time_level = 'MONTH' BusinessObjects Universe Builder Guide 41 6 Building universes from Oracle Analytic Services

How are Oracle OLAP structures mapped to universe components? Joined tables Expression QUARTER, OLAPCUBE /* QUARTER.DUMMY */ OLAPCUBE.time_level = 'QTR' /* YEAR.DUMMY */ OLAPCUBE.time_level = 'YEAR' /* TIME_ALL.DUMMY */ OLAPCUBE.time_level = 'ALL' YEAR, OLAPCUBE TIME_ALL, OLAPCUBE How are Oracle OLAP structures mapped to universe components? To obtain and set up the expected universe, the Oracle OLAP Universe Builder adds and configures universe objects as follows: View Oracle OLAP Universe Builder inserts the relational view as a table in the universe as well as the oracle table sys.dual.

If you choose to use a derived table, a derived table is inserted with the definition of the view (select part with OLAP_TABLE function). Hierarchy tables For each hierarchy that is represented in the relational view, an alias of sys.dual is created for each level of the hierarchy. The name of the alias is the level name.

For example: if we have a dimension TIME with 4 levels (ALL, YEAR, MONTH, QUARTER) then create 4 aliases ALL, YEAR, MONTH, QUARTER. 42 BusinessObjects Universe Builder Guide Building universes from Oracle Analytic Services How are Oracle OLAP structures mapped to universe components? 6 Multi-hierarchy tables Note: Multi-Hierarchy is a special case. See the section Multi-Hierarchy special case: (Multi-Hierarchy dimensions support) on page 46 for more information. If a dimension has more than one hierarchy, then a different set of tables is created for each hierarchy even if some hierarchies share a same level. This means that for shared levels, as many aliases are created as hierarchies. The naming of such aliases is a concatenation of the level name and the hierarchy name.

For example: Dimension time has two hierarchies: H1 (All_Time, Year, Month) and H2 (All_Time, Quarter, Month) All_Time and Month are shared by both hierarchies so we'll have two aliases for All_Time: All_Time_H1 and All_Time_H2 And two aliases for Month: Month_H1 and Month_H2 Dimension joins - Each table representing a level is joined to its direct lower level in the same hierarchy.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS UNIVERSE BUILDER user guide](http://yourpdfguides.com/dref/3796642)

<http://yourpdfguides.com/dref/3796642>

The join expression is: `/* Alias1.DUMMY=Alias2.DUMMY */ I=1` BusinessObjects Universe Builder Guide 43 6 Building universes from Oracle Analytic Services How are Oracle OLAP structures mapped to universe components? where Alias1 represents a level and Alias2 represents its direct upper level in the hierarchy. Example: `/* Quarter.`

`DUMMY=Year.DUMMY */ I=1` · Each table is joined to the view using a shortcut join type except for the lowest level where the join is regular. The join expression defines a value to filter the rows returned from the view and is of the type: `/* Alias.DUMMY */ VIEW.levelColumn = 'level_value'` Where Alias is the alias name, levelColumn is the column representing the level within the view and level_value is the value of that column that matches the level name.

Examples: MYVIEW is the view that represents the OLAP cube, the column that contains the levels is time_level, the level values are: ALL, YEAR, QTR, MONTH. Joined tables expression `MONTH, MYVIEW /* MONTH.DUMMY */ MYVIEW.time_level = 'MONTH' QUARTER, MYVIEW /* QUARTER.DUMMY */ MYVIEW.time_level = 'QTR' YEAR, MYVIEW /* YEAR.DUMMY */ MYVIEW.time_level = 'YEAR' TIME_ALL, MYVIEW /* TIME_ALL.DUMMY */ MYVIEW.time_level = 'ALL'` Classes and Objects The Bridge must create a class for each OLAP dimension and an object for each level. Class and object properties are mapped as follows: Mapped from OLAP item.

.. Universe item Property Class See also Multi-Hierarchy special case: (Multi-Hier- Name archy dimensions support) on page 46 below. Dimension name 44 BusinessObjects Universe Builder Guide Building universes from Oracle Analytic Services How are Oracle OLAP structures mapped to universe components? 6 Universe item Property Mapped from OLAP item... Objects Levels and attributes. Current level name identified by the field name in the view. View field, for example MYVIEW.YEAR.

Additional tables to ensure the use of appropriate joins: · Table of current view · Dimension Name Select Tables View Highest level table for all dimensions Detail (optional) Dimension object created from the Description field that is related to the Parent dimension ID field. For example, ID Note: All other proper-field is YEAR, Descripties are the same as for tion field it YEAR_DEIDimension above. SC. The object YEAR is a detail of object YEAR_DESC. Name Measure name in cube.

Measure BusinessObjects Universe Builder Guide 45 6 Building universes from Oracle Analytic Services How are Oracle OLAP structures mapped to universe components? Universe item Property Mapped from OLAP item... Select (no drill through) View field, for example MYVIEW.SALES Additional tables to ensure the use of appropriate joins: · View · Tables Highest level table for all dimensions Aggregation function None.

Multi-Hierarchy special case: (Multi-Hierarchy dimensions support) To support Multi-hierarchy dimensions of an Analytical Warehouse, the following actions are carried out in the universe: · A set of tables is created for each hierarchy as described in the section Multi-hierarchy tables on page 43. A class is created for the dimension and a subclass for each hierarchy. The naming of subclasses is a concatenation of the dimension name and the hierarchy name. For each object corresponding to a level within a hierarchy, the aggregate function is added to the Select expression. It references as comments, all the high level aliases, except the aliases of the current hierarchy. For example: · 46 BusinessObjects Universe Builder Guide Building universes from Oracle Analytic Services How are Oracle OLAP structures mapped to universe components? 6 · Aggregate Navigation is set to make the objects of a subclass (Hierarchy) incompatible with the tables corresponding to another hierarchy. This prevents the end user from using in a report objects representing levels that belong to different hierarchies. For example, the table Year_H1 (from Hierarchy H1) is incompatible with objects from H2 hierarchy: And the table Year_H2 (from Hierarchy H2) is incompatible with objects from H1 hierarchy: BusinessObjects Universe Builder Guide 47 6 Building universes from Oracle Analytic Services How to create a Oracle OLAP universe How to create a Oracle OLAP universe You can create universes and views as using the following creation methods: Universe creation option Description Creating a view and generating a universe on page 49 You create a view, and then select which structures are mapped to create the universe. You can create a view and save the view if you do not want to create a universe. The view is available in a view list and can be used to create a universe at any time.

Create a view only on page 51 You select an existing view, and Creating Oracle OLAP universe from generate a universe directly from this an existing view on page 52 view. 48 BusinessObjects Universe Builder Guide Building universes from Oracle Analytic Services How to create a Oracle OLAP universe 6 Creating a view and generating a universe You generate an Oracle OLAP universe by first defining a view using Analytic Workspace Cube metadata, then you set universe creation options, and generate a new universe. 1. Click the Oracle OLAP icon or tab. 2. Select File > Metadata Sources. The Oracle OLAP Connection Wizard Welcome page appears. 3. Click Next. The wizard login page appears.

4. Select a connection, enter user name and password, and then click Finish. The Universe Builder Panel appears displaying the Analytic Workspace Cube cubes available to the connection 5. Click the Cubes node. The Analytic Workspaces (AW) available in the connection are displayed. 6. Expand a AW node to display the cubes available to the AW, then right click a cube and select Create Universe from the context menu. The Universe Builder wizard appears. Three options are available. These are described in the section How to create a Oracle OLAP universe on page 48. 7. Click the Create View and generate Universe radio button, then click Next. A status box appears showing the metadata loading status. A box showing data type and data length information appears. 8. If required, modify the data type and length values. Do this as follows: · Double click a data type or length value. · Select a data type from the drop down list box. · Click Next. The Hierarchy level page appears.

This page lists the hierarchy levels with data types and values. 9. Edit hierarchy values if necessary, then click Next. The View and universe options page appears. BusinessObjects Universe Builder Guide 49 6 Building universes from Oracle Analytic Services How to create a Oracle OLAP universe 10. Type a name for the view, and select view and universe options. The View properties and universe options are as follows: Options Description View Name Name for the view.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS UNIVERSE BUILDER user guide](http://yourpdfguides.com/dref/3796642)
<http://yourpdfguides.com/dref/3796642>

You can edit this field. When selected, an extra column of type Raw(32) is created in the view Create Column for OLAP_EXPRESto enable the use of OLAP_EXSION PRESSION function in the universe. When selected, columns representing the dimension members (identifiers) are created. Create Columns for Identifiers When selected existing Type and Replace existing database objects View structures are replaced in the database. When selected the universe is not built from a view physically created in the database, but is built by using a derived table to reference the cube structures. A derived table is a virtual table that exists only in the universe that references database structures. It is not created in the database. @@@@Click Next. The SQL verification page appears. 12. Verify the SQL for the view and click Next. The universe information summary page appears. 13. Verify the universe information, and click Finish. @@The saved view appears in a list of views. @@To create a view only, follow the same procedure in the section Creating a view and generating a universe on page 49Creating a view and generating a universe on page 49 but at the start of the Oracle OLAP Universe Builder wizard, you select the Create View only radio button. The view is created in the target database. You can connect to this view at any time to create a universe. See the section Creating Oracle OLAP universe from an existing view on page 52for the procedure on using a view to create a universe. BusinessObjects Universe Builder Guide 51 6 Building universes from Oracle Analytic Services How to create a Oracle OLAP universe Creating Oracle OLAP universe from an existing view You can generate a universe from an existing view. Existing views appear in a list. You select a view from the list and generate a universe. 1.

Click the Oracle OLAP icon or tab. 2. Select File > Metadata Sources. The Oracle OLAP Connection Wizard Welcome page appears. 3. Click Next. The wizard login page appears. 4. Select a connection and enter user name and password. 5. Click Finish. The Universe Builder Panel appears displaying the Analytic Workspace Cube cubes available to the connection. 6. Click the Cubes node. The Analytic Workspaces (AW) available in the connection are displayed. Expand a AW node to display the cubes available to the AW. 7. Right click a cube and select Create Universe from the context menu. The Universe Builder wizard appears. Three options are available. How to create a Oracle OLAP universe on page 48 for a description of each option. 8. Click the Generate Universe from View radio button. 9. Click Next. A list of available views defined on the cube appears. 10. Click a view name in the list and click Next. A status box shows the progress of metadata being loaded from the selected cube. The Universe Creation page appears.

It lists dimension, measures, and hierarchy levels defined in the view that can be used to create a universe. 11. If required, modify a column name or hierarchy level. Do this as follows: · Double click a column name or level value. · Select or type a name as appropriate. 12. Click Finish. Designer starts and opens to the newly generated universe. 52 BusinessObjects Universe Builder Guide Get More Help A A Get More Help Online documentation library Business Objects offers a full documentation set covering all products and their deployment. The online documentation library has the most up-to-date version of the Business Objects product documentation. You can browse the library contents, do full-text searches, read guides on line, and download PDF versions. The library is updated regularly with new content as it becomes available. http://support.businessobjects.com/documentation/product_guides/ Additional developer resources <http://devlibrary.businessobjects.com> Online customer support The Business Objects Customer Support web site contains information about Customer Support programs and services. It also has links to a wide range of technical information including knowledgebase articles, downloads, and support forums. <http://www.businessobjects.com/support/> Looking for the best deployment solution for your company? Business Objects consultants can accompany you from the initial analysis stage to the delivery of your deployment project. Expertise is available in relational and multidimensional databases, in connectivities, database design tools, customized embedding technology, and more. For more information, contact your local sales office, or contact us at: <http://www.businessobjects.com/services/consulting/> Looking for training options? From traditional classroom learning to targeted e-learning seminars, we can offer a training package to suit your learning needs and preferred learning style. Find more information on the Business Objects Education web site: <http://www.businessobjects.com/services/training> 54 BusinessObjects Universe Builder Guide Get More Help A Send us your feedback Do you have a suggestion on how we can improve our documentation? Is there something you particularly like or have found useful? Drop us a line, and we will do our best to ensure that your suggestion is included in the next release of our documentation: <mailto:documentation@businessobjects.com> Note: If your issue concerns a Business Objects product and not the documentation, please contact our Customer Support experts. For information about Customer Support visit: <http://www.businessobjects.com/support/>. Business Objects product information For information about the full range of Business Objects products, visit: <http://www.businessobjects.com>. BusinessObjects Universe Builder Guide 55 A Get More Help 56 BusinessObjects Universe Builder Guide Index C classes and objects in Oracle Analytic Workspaces universe 42 O Oracle Analytic Workspaces class and object mapping 42 dimension joins 42 hierarchy tables 42 multi-hierarchy tables 42 use of shortcut joins in universe 40 view 42 D dimension joins use in Oracle Analytic Workspaces universe 42 S H hierarchy tables use in Oracle Analytic Workspaces universe 42 shortcut join use in Oracle Analytic Workspaces universe 40 V M multi-hierarchy tables use in Oracle Analytic Workspaces universe 42 view use in Oracle Analytic Workspaces universe 42 BusinessObjects Universe Builder Guide 57 Index 58 BusinessObjects Universe Builder Guide .



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS UNIVERSE BUILDER user guide](http://yourpdfguides.com/dref/3796642)
<http://yourpdfguides.com/dref/3796642>