Oracle® Utilities Work and Asset Management

Business Intelligence Guide

Release 1.9.0.3

July 2011

ORACLE®

Oracle® Utilities Work and Asset Management Business Intelligence Guide for Release 1.9.0.3

Doc v1 rev.0 7/7/11

Business Intelligence

Oracle Utilities Business Intelligence (BI) for Oracle Utilities Work and Asset Management uses extraction programs packaged with the Oracle Utilities Application Framework (OUAF). The extraction programs for Oracle Utilities Work and Asset Management are written in PL/SQL and are loaded into BI using OUAF.

Users can view batch control records, submit batch jobs and view the batch run tree using the Oracle Utilities Work & Asset Management application.

This section describes the configurations required in Oracle Utilities Work and Asset Management only. For complete Oracle Utilities Business Intelligence implementation and configuration guidelines refer to the online documentation within BI.

Note: It is recommended that you read this entire document before beginning installation.

Certified Version

Certified versions of Business Intelligence include:

- Business Intelligence v2.2.1 and higher for Oracle Utilities Work and Asset Management releases 1.8 and higher.

- Business Intelligence v2.3.0 for Oracle Utilities Work and Asset Management releases 1.9.0.2 and above.

Extract Mapping

Please refer to the printed User Guide for extract mapping information.

Topics

Basic Steps to Get Started
Multiple Plants
Extract Output
Batch Procedure
Business Rules
Sequence Numbering
Initial Load and Incremental Load
Error Handling
File Name Conventions and Outputs
SABI Triggers
Extracts

Basic Steps to Get Started

1. **Enable the Oracle Utilities Work and Asset Management BI database triggers using Installation Parameters.**
2. **Setup the BI Batch procedure, SDBP_BI_EXTRACT_HANDLER, in the Job Manager module.**
3. **Configure the OUBI Setup Business Rule.**

4. **Verify sequence numbering for SA_CHANGE_LOG in the Sequence Numbering module.**
5. **Run the batch procedure, SDBP_BI_EXTRACT_HANDLER.**

Multiple Plants

If you have multiple plants all configuration settings must be configured and executed separately for each.

Extract Output

A folder for the extraction file output is required to be created on the database server using the database server operating system. This directory is referred to as the "BI Extract output folder". A database object called a "Directory" needs to be created to point to this folder on the database server file system.

Default
The installation is includes a pre-configured database directory object that is defaulted to "**c:\bi_extract**". This serves as a target folder for extracted files. **SD_BI_OUTBOUND** is the directory object in the database that maps to the output directory folder for the BI extracts.

Override the Default
This default can be overridden at the database level by running the following script:

```
CREATE OR REPLACE DIRECTORY sd_bi_outbound AS '<operating system folder
name>'
/
```

Steps to Create the Database Object
1. **The Directory object must be created by a user with DBA privileges.**
2. **Create the directory object referencing your output folder.**
   To override the default, run the script indicated above.

3. **Grant privileges to the Directory to the user that will be executing the extract routines. This is the same user that will be creating the objects in the Oracle Utilities Work and Asset Management Job Scheduler.**
   grant read on directory SD_BI_OUTBOUND to synergen;

   grant write on directory SD_BI_OUTBOUND to synergen;

Batch Procedure

The batch procedure, SDBP_BI_EXTRACT_HANDLER, handles and calls all of the extract programs. You can monitor this processing in the Job Manager and the Job Manager Log modules. SDBP_BI_EXTRACT_HANDLER also handles incrementing the value of the sequence number whenever an incremental extract is requested.

Business Rules
OUBI Setup Rule

Configure settings to determine which extracts will be completed during regular processing. This rule can also be used to define extraction details such as data source, file names and the output directory. If desired, any extract can be turned OFF so that it is not included in the overall extraction.

Common Extract Parameters
The first rule key, COMMON EXTRACT PARAMETERS, defines parameters for all of the extracts that are turned ON. The key values for this rule key differ slightly from the key values for the remaining rule keys.

Value is left blank for this rule key.

Value2 is the name of the directory object in the database. This corresponds to the output folder of the extract, SD_BI_OUTBOUND.

Value3 is the Data Source Indicator. Default value is '3'

Value4 controls which type of extract will be executed. A value of INI means the extract will produce an Initial Extract, an INC means an Incremental Extract. The default value is INI.

After an Initial Extract, the value is automatically updated to INC by SDBP_BI_EXTRACT_HANDLER so that the succeeding extracts are incremental.

General Key Names and Values
Value controls whether the extract is executed. Default value is ON.

Value2 is the base extract file name. This is the extract identifier used in building the names of the extract files.

Value3 and Value4 only apply to the Key Name OUBI INV SNAPSHOT. The goal of this metric is to compare inventory levels of like stock items based on periodic snapshots of the quantities in the storeroom. This is not a metric used to compare inventory movement because it does not deal with the transaction volume.

This requires two parameters to be passed to the procedures:

Value3 - Type of Increment - Monthly or Daily

Value4 - Number of Historical Snapshots - Any number

Sequence Numbering
Sequence Numbering must be turned on for SA_CHANGE_LOG to control the batch number field of the SA_CHANGE_LOG table. The Oracle Utilities Business Intelligence triggers use the value of this sequence number to uniquely identify records inserted into the SA_CHANGE_LOG table.

Initial Load and Incremental Load
Settings for the OUBI Setup Business Rule COMMON EXTRACT PARAMETERS rule key determine how the data is initially loaded as well as the incremental data loads.

Initial Load
The Initial Load of the extracts will query ALL records from the target tables and sends the entire set of source data to BI. This load is filtered by PLANT. There is no additional filtering applied to the source information. For example, the Account Transactions Extract targets the SA_ACCOUNT_LOG table as the source information. The initial load will take all records in the SA_ACCOUNT_LOG table and send the records to BI.

If you need to extract records from another plant, the batch job, SDBP_BI_EXTRACT_HANDLER, must be for the other PLANT and executed. Same with the Incremental extracts.

Incremental Load
The Incremental Load is based on information found in the SA_CHANGE_LOG table. Records are inserted into this table from a set of BI extract database triggers placed on various tables in Oracle Utilities Work and Asset Management. The incremental extracts begin to have data in the files after the BI triggers have been enabled in the Oracle Utilities Work and Asset Management database schema. If a record is recorded in the SA_CHANGE_LOG, a transaction is set to BI.

Error Handling
Errors will not occur unless there is a problem with the extracted data or with configuration settings. If errors are encountered, they will be noted in the Job Manager Log module. Correct any errors then re-run an Initial Load.

To Rerun an Initial Load
1. **Open the OUBI Setup Business Rule.**
2. **Set the Common Extract Parameters rule key to 'INI.'**
3. **Execute the batch job, SDBP_BI_EXTRACT_HANDLER.**
   After an Initial Extract, the business rule value is automatically updated to INC by SDBP_BI_EXTRACT_HANDLER so that the succeeding extracts are incremental.

File Name Conventions and Outputs

File Names follow the standard naming convention expected by OUBI. The OUBI Setup Business Rule handles the prefix file name for all the extracts. This is the value under the Value2 column. These filenames cannot be modified. The structure is indicated here to provide insight into how information can be determined from the file name.

Example: EXTDASSE0000030000000021001.<extension name>

- Extract Identifier: No limit on the size

- Data Source Indicator: Next 6 characters

- Batch Number: Next 10 characters

- Thread Number: Next 3 characters

Each extract produces two files: *.DAT and *.CTL.

- **DAT** files contain the actual extract data from the Oracle Utilities Work and Asset Management tables.

- **CTL** files are used by Oracle Utilities BI for log and upload trace details. The file contains one line of information which consists of:

  - Batch Code: First 8 characters
  - Data Source Indicator: Next 6 characters
  - Batch Number: Next 10 characters
  - Thread Number: Next 10 characters
  - Line Count (on the DAT file): Next 10 characters
  - Batch Code Description: Next 60 characters

SABI Triggers

SABI triggers insert data into the SA_CHANGE_LOG table in preparation for extraction. The triggers use sequence number settings to determine the next batch number for the extraction.

The following triggers are used:

bi_adiu_acct_data	bi_adiu_actg_log
bi_adiu_actperdates	bi_adiu_asset
bi_adiu_assetdown	bi_adiu_auth_stt
bi_adiu_catalog	bi_adiu_dept
bi_adiu_dept_oa	bi_adiu_failure
bi_adiu_op_actg_tr_ty	bi_adiu_ouom
bi_adiu_repair	bi_adiu_rootcause
bi_adiu_rule_key_oex	bi_adiu_rulekey_stt
bi_adiu_storeroom	bi_adiu_strm_setup
bi_adiu_strm_trans	bi_adiu_tracwrkordtcr
bi_adiu_tracwrkordtpl	bi_adiu_wrkord

bi_adiu_acct_data	bi_adiu_actg_log
bi_adiu_wrkordadjust	bi_adiu_wrkordlbr
bi_adiu_wrkordmat	bi_adiu_wrkordrequire
bi_adiu_wrkordservcont	bi_adiu_wrkordtask
bi_adiu_wrkordtfail	bi_adiu_wrkordty

Extracts

The following section maps the extract program to the data that it handles as well as the view, batch procedure and trigger that are used to complete the extraction.

Please refer to the extract mapping document titled, WAM BI Extract Mapping for more information.

To Review the Columns that are Extracted

Access the database and describe the definition of the view. From there you can see the specific columns that comprise the view.

For example, in SQL Plus:

```
SELECT
 *
FROM SYNERGEN.SV_BI_EXTRACT_EXTDFAIL Vw
```