Oracle Utilities Analytics

Administration Guide

Release 2.7.0.1

F19322-01

July 2019

ORACLE®

Oracle Utilities Analytics Administration Guide, Release 2.7.0.1

F19322-01

Contents

Chapter 3

Chapter 4

Chapter 5

Chapter 6

Preface

Welcome to the Oracle Utilities Analytics Administration Guide.

This guide provides instructions for configuring and administering Oracle Utilities Analytics. The preface focuses on the following:

- Audience

- Prerequisite Knowledge

- How This Guide is Organized

- Related Documents

- Conventions

- Documentation Accessibility

- Acronyms

- Terminology

Audience

This guide is intended for system and data warehouse administrators who need to configure and administer Oracle Utilities Analytics. Implementers and developers may also find it useful in understanding how the application works.

Prerequisite Knowledge

Oracle Utilities Extractors and Schema and Oracle Utilities Analytics Dashboards make use of several technologies. It is assumed that you have a working knowledge of the following to configure and administer Oracle Utilities Analytics:

- Oracle Data Warehouse

    https://docs.oracle.com/en/database/oracle/oracle-database/18/dwhsg/index.html

- Oracle GoldenGate

    https://docs.oracle.com/middleware/12213/cross/getstartedtasks.htm

- Oracle Data Integrator

    http://docs.oracle.com/goldengate/1221/cross/getstartedtasks.htm

- Oracle GoldenGate Monitor

    https://docs.oracle.com/goldengate/m12212/gg-monitor/index.html

- Oracle WebLogic Server

    https://docs.oracle.com/middleware/12213/wls/index.html

- Oracle Business Intelligence Enterprise Edition

    https://docs.oracle.com/middleware/12213/bisuite/docs.htm

How This Guide is Organized

This guide helps you configure and administer Oracle Utilities Extractors and Schema and Oracle Utilities Analytics Dashboards. This guide refers to these two products together as Oracle Utilities Analytics. If any topic is specific to only one of the products, then it is specifically mentioned.

Chapter 1: Overview of Oracle Utilities Analytics introduces the reader to Oracle Utilities Analytics product, its fundamental architecture and how the product works.

Chapter 2: Oracle Utilities Extractors and Schema lists the various components related to the extractors and schemas within the product and how they work together in the product.

Chapter 3: Oracle Utilities Analytics Dashboards lists the various components related to the dashboards within the product and how they work together in the product.

Chapter 4: Oracle Utilities Analytics Administration Tool describes how to maintain the out-of-the-box ELT jobs using the Oracle Utilities Analytics Administration Tool.

Chapter 5: Configuring Oracle Utilities Analytics goes into detail on how to configure the Oracle Utilities Analytics product and how to manage and maintain an implementation of Oracle Utilities Analytics.

Chapter 6: Managing Oracle Utilities Analytics delves into the important configurations related to the Oracle Utilities Analytics product and additional recommendations on utilizing the partitioning feature of the Oracle database.

Related Documents

The following documentation is included with this release.

Installation, Administration, and Release Notes

- *Oracle Utilities Analytics Release Notes*

- *Oracle Utilities Analytics Getting Started Guide*

- *Oracle Utilities Analytics License Information User Manual*

- *Oracle Utilities Analytics Installation Guide*

- *Oracle Utilities Analytics Quick Install Guide*

- *Oracle Utilities Analytics Administration Guide*

- *Oracle Utilities Analytics Developer's Guide*

Metric Reference Guides

- *Oracle Utilities Analytics Dashboards for Oracle Utilities Meter Data Analytics Metric Reference Guide*

- *Oracle Utilities Analytics Dashboards for Oracle Utilities Customer Analytics, Revenue Analytics and Credit & Collections Analytics Metric Reference Guide*

- *Oracle Utilities Analytics Dashboards for Oracle Utilities Exception Analytics Metric Reference Guide*

- *Oracle Utilities Analytics Dashboards for Oracle Utilities Mobile Workforce Analytics Metric Reference Guide*

- *Oracle Utilities Analytics Dashboards for Oracle Utilities Distribution Analytics and Outage Analytics Metric Reference Guide*

- *Oracle Utilities Analytics Dashboards for Oracle Utilities Work and Asset Analytics Metric Reference Guide*

- *Oracle Utilities Analytics Dashboards for Oracle Utilities Operational Device Analytics Metric Reference Guide*

Data Mapping Guides

- *Oracle Utilities Extractors and Schema for Oracle Utilities Customer Care and Billing Data Mapping Guide*

- *Oracle Utilities Extractors and Schema for Oracle Utilities Meter Data Management Data Mapping Guide*

- *Oracle Utilities Extractors and Schema for Oracle Utilities Mobile Workforce Management Data Mapping Guide*

- *Oracle Utilities Extractors and Schema for Oracle Utilities Network Management System Data Mapping Guide*

- *Oracle Utilities Extractors and Schema for Oracle Utilities Operational Device Management Data Mapping Guide*

- *Oracle Utilities Extractors and Schema for Oracle Utilities Work & Asset Management Data Mapping Guide*

You can view the latest documentation at http://docs.oracle.com/cd/E72219_01/documentation.html.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Documentation Accessibility

For information about configuring and using accessibility features for Oracle Utilities Analytics, see the documentation at http://docs.oracle.com/cd/E23943_01/bi.1111/e10544/appaccess.htm#BIEUG2756.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/us/corporate/accessibility/index.html.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For more information, visit: http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Acronyms

The list of acronyms used in this guide is as below:

- **APEX**: Oracle Application Express
- **CC&B**: Oracle Utilities Customer Care and Billing
- **CDC**: Changed Data Capture
- **ELT**: Extraction, Loading, and Transformation
- **ETL**: Extraction, Transformation, and Loading
- **MDM**: Oracle Utilities Meter Data Management
- **MWM**: Oracle Utilities Mobile Workforce Management
- **NMS**: Oracle Utilities Network Management System
- **OBIEE**: Oracle Business Intelligence Enterprise Edition
- **ODI**: Oracle Data Integrator
- **ODM**: Oracle Utilities Operational Device Management
- **OGG**: Oracle GoldenGate
- **OUA**: Oracle Utilities Analytics
- **OUAF:** Oracle Utilities Application Framework

- **OWB**: Oracle Warehouse Builder

- **WAM**: Oracle Utilities Work and Asset Management

Terminology

This section covers basic terminology related to Oracle Utilities Analytics and the other products needed for its implementation:

- Oracle Utilities Analytics

- Oracle Data Integrator

- Oracle GoldenGate

- Oracle Business Intelligence Enterprise Edition

Oracle Utilities Analytics

This sub-section includes the following:

- Data Source Indicator

- Entity

- Instance

- Late Arriving Dimension

- Product

Data Source Indicator

The data warehouse is an integrated database receiving data from the multiple source applications. The Data Source Indicator (DSI) in Oracle Utilities Analytics is used as an identifier for the originating source application or system-of-record for the data. Both fact and dimension tables have a DSI value for each record.

Entity

The data warehouse is built of star schemas that consist of facts and dimensions. Each fact and dimension is considered to be an entity. We also use materialized views to aggregate data in the facts for improving the analytics performance; these are also considered entities.

Instance

Each deployment of the source product is considered to be an instance of that product. A product instance may include multiple servers in a clustered deployment. Instances differ by the source database connections and possibly by the differing configurations. The Data Source Indicator (DSI) is unique across instances and products as this allows the ability to identify the source of the data in the data warehouse.

Late Arriving Dimension

The late arriving dimensions are the dimensions where the fact (measurable quantities) table records come early when compared to the dimension table records.

Product

Product is the source system or edge application that act as the source for the transactional data that is loaded in the data warehouse. The following scenarios are expected depending on your global presence and the implementation of your products:

- The edge applications may be deployed in different locations across the globe and, thus, the transactions may occur in multiple time zones.

- There may be multiple instances of each of the source systems that you have implemented.
- Oracle Utilities products support cross-product integration and, therefore, it is possible that some of the master data is integrated across the multiple systems.

Oracle Data Integrator

This sub-section includes the following:

- Agent
- Change Data Capture
- Context
- Interface
- Journalization
- Knowledge Modules
- Logical Schema
- Model
- Package
- Physical Schema
- Reverse Engineering
- Scenario

Agent

An agent is a component that executes the Oracle Data Integrator jobs. Multiple agents can be created and utilized for load balancing.

Change Data Capture

It is the process of identifying and tracking data that has changed since the last known state. The change data capture mechanisms are implemented by the Journalizing Knowledge Modules (JKM).

Context

A context is a collection of logical schemas mapped to physical schemas. All logical schemas may not be mapped to the physical schemas in a context. A context identifies the physical schemas and connections that are utilized for the data access. The interfaces can be executed in any context as this allows reusing the same code for multiple instances.

Interface

An interface is a declarative mapping between the source to the target. It utilizes Loading Knowledge Modules (LKM), Integration Knowledge Module (IKM), and Check Knowledge Module (CKM), as required, for execution. The joins, aggregations, filters, and data transformations are performed in an interface.

Journalization

It is the process of keeping track of data changes, enabling incremental data access, transport and transformation.

Knowledge Modules

Oracle Data Integrator Knowledge Modules (KMs) are components that implement reusable transformation and ELT (Extraction, Loading, and Transformation) strategies across different technologies.

- Knowledge modules are code templates. Each KM is dedicated to an individual task in the overall data integration process.

- Knowledge modules may be reused across several interfaces or models so a change made to a KM will be automatically propagated to all of the transformations that include it.

- Knowledge modules are based on logical tasks that are performed; they do not contain references to the physical objects (data stores, columns, physical paths, *etc.*).

- Knowledge modules may be written with different programming languages, types, and styles (*e.g.*, native RDBMS SQL, scripting languages such as Jython or JavaScript, or even Java).
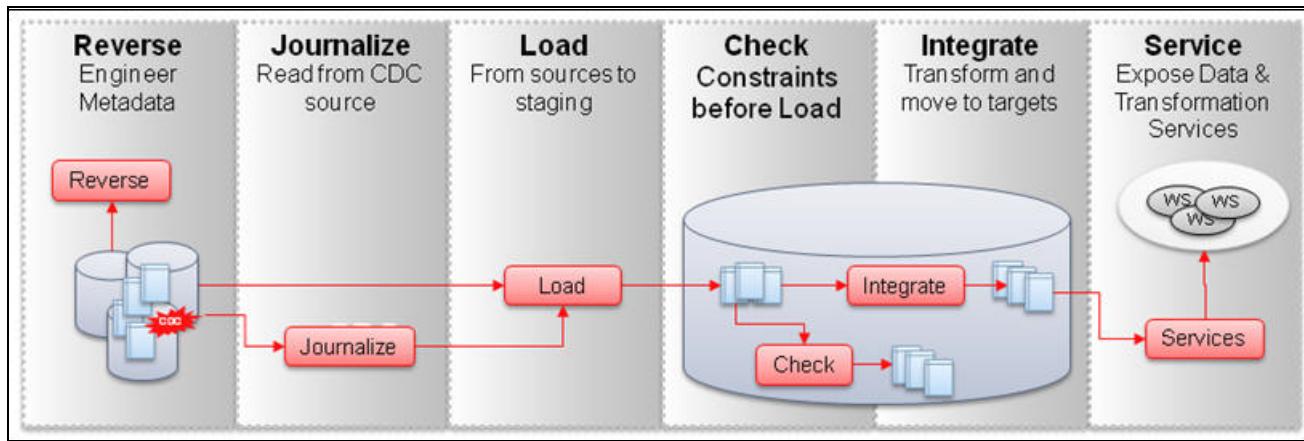
Types of Knowledge Modules

The types of knowledge modules are:

1. Reverse-engineering Knowledge Modules (RKM)

2. Check Knowledge Modules (CKM)

3. Loading Knowledge Modules (LKM)

4. Integration Knowledge Modules (IKM)

5. Journalizing Knowledge Modules (JKM)

6. Service Knowledge Modules (SKM)

> **Note**: Refer to Oracle Data Integrator documents for more details.

Logical Schema

A logical schema is a logical representation of a storage location. The same physical schemas can be associated with different logical schemas.

Model

A model is a set of data stores corresponding to data structures contained in a physical schema. The tables, files, JMS messages, elements from an XML file are represented as data stores.

Package

A package is made up of a sequence of steps organized into an execution diagram.

Physical Schema

A physical schema is the actual schema where data is retrieved or stored.

Reverse Engineering

Reverse engineering is the process of identifying the data structures, constraints, and relations from an existing data store. The data store can be any database management system, files, or any other Oracle Data Integrator supported data store.

Scenario

A scenario is a self-contained prepackaged executable version of a package, interface, or procedure in Oracle Data Integrator. The logic within a scenario cannot be viewed using the Oracle Data Integrator client.

Oracle GoldenGate

This sub-section includes the following:

- Data Pump
- Extract
- Manager
- Parameter files
- Replicat
- Trail Files

Data Pump

A data pump is a secondary extract group within the source Oracle GoldenGate configuration. If a data pump is not used, the Extract must send the captured data operations to a remote trail on the target. In a typical configuration with a data pump, however, the primary Extract group writes to a trail on the source system. The data pump reads this trail and sends the data operations over the network to a remote trail on the target. The data pump adds storage flexibility and also serves to isolate the primary extract process from the TCP/IP activity.

In general, a data pump can perform data filtering, mapping, and conversion, or it can be configured in pass-through mode, where data is passively transferred as-is, without manipulation. The pass-through mode increases the throughput of the data pump. This bypasses all object definition lookups.

Extract

The Extract process is the extraction (capture) mechanism of Oracle GoldenGate. The extract process runs on a source system or on a downstream database, or both, depending on the database and the implementation requirements.

When configured for change synchronization, extract process captures the DML and DDL operations that are performed on objects in the Extract configuration. The Extract process stores these operations until it receives commit records or rollbacks for the transactions that contain them. When a rollback is received, the Extract discards the operations for that transaction. When a commit is received, the Extract persists the transaction to disk in a series of files called a trail, where it is queued for propagation to the target system. All of the operations in each transaction are written to the trail as a sequentially organized transaction unit. This design ensures both speed and data integrity.

Manager

The Manager process performs a number of functions such as starting other Oracle GoldenGate processes, trail log file management and reporting. The Manager process needs to be configured on both source as well as target systems and configuration is carried out via a parameter file. This is the mgr.prm file.

Parameter files

Most Oracle GoldenGate functionalities are controlled by means of parameters specified in the parameter files. A parameter file is a plain text file that is read by an associated Oracle GoldenGate process.

Replicat

The Replicat process runs on the target system, reads the trail on that system, and then reconstructs the DML or DDL operations and applies them to the target database. The Replicat uses the dynamic SQL to compile a SQL statement once, and then execute it many times with different bind variables.

Trail Files

To support the continuous extraction and replication of the database changes, Oracle GoldenGate stores records of the captured changes temporarily on the disk in a series of files called a trail. A trail can exist on the source system, an intermediary system, the target system, or any combination of those systems, depending on how you configure Oracle GoldenGate. On the local system, it is known as an extract trail (or local trail). On a remote system, it is known as a remote trail.

Oracle Business Intelligence Enterprise Edition

This section defines the certain key terms for Oracle Business Intelligence Enterprise Edition Tool used in the Oracle Utilities Analytics product, including:

- Analysis/Report

- Business Model and Mapping Layer

- Dashboard

- Oracle Business Intelligence Repository

- Oracle Business Intelligence Presentation Catalog

- Physical Layer

- Presentation Layer

- Subject Area

> **Note**: For an exhaustive list, refer to the product documentation of Oracle Business Intelligence Enterprise Edition.

Analysis/Report

This is a query that you create on the **Criteria** tab in **Presentation Services.** An analysis can optionally contain one or more filters or selection steps to restrict the results.

Business Model and Mapping Layer

The Business Model and Mapping Layer defines the business or logical model of the data and specifies the mapping between the business model and the physical layer schemas. This layer can contain one or more business models. The business model and mapping layer determines the analytic behavior that is seen by users, and defines the superset of objects available to you. It also hides the complexity of the source data models. Sometimes, this layer is also referred to as the logical layer.

Dashboard

An object that provides personalized views of corporate and external information. A dashboard consists of one or more pages. Pages can display anything that you can access or open with a web browser, such as results of analyses, images, alerts from agents, and so on.

Oracle Business Intelligence Presentation Catalog

It stores business intelligence objects, such as analyses and dashboards, and provides an interface where you create, access, and manage objects, and perform specific object-based tasks (for example, export, print, and edit). The catalog is organized into folders that are either shared or personal.

Oracle Business Intelligence Repository

It is a file that stores Oracle Business Intelligence metadata. The metadata defines logical schemas, physical schemas, physical-to-logical mappings, aggregate table navigation, and other constructs. The repository file has an extension of .rpd. Oracle Business Intelligence repositories can be edited using the Oracle Utilities Analytics Administration Tool.

Physical Layer

It is a layer of the Oracle Business Intelligence repository that contains objects that represent physical data constructs from back-end data sources. The physical layer defines the objects and relationships available for writing physical queries. This layer encapsulates source dependencies to enable portability and federation.

Presentation Layer

This layer provides a way to present customized, secure, role-based views of a business model to you. It adds a level of abstraction over the business model and the mapping layer in the Oracle Business Intelligence repository. The presentation layer provides the view of the data seen by you who build analyses in **Presentation Services** and other client tools and applications.

Subject Area

In an Oracle Business Intelligence repository, an object in the presentation layer that organizes and presents data about a business model. It is the highest-level object in the presentation layer and represents the view of the data that you see in **Presentation Services.** Oracle Business Intelligence repository subject areas contain presentation tables, presentation columns, and presentation hierarchies. In **Presentation Services**, subject areas contain folders, measure columns, attribute columns, hierarchical columns, and levels.

Chapter 1

Overview of Oracle Utilities Analytics

This chapter provides an overview of Oracle Utilities Analytics.

- Introduction

- Architecture

Introduction

Oracle Utilities Analytics is a suite of pre-built analytics applications for the utilities industry. It consists of a single data repository with available data integrations to the following Oracle Utilities Source (Edge) Applications:

- Oracle Utilities Customer Care and Billing
- Oracle Utilities Meter Data Management
- Oracle Utilities Mobile Workforce Management
- Oracle Utilities Network Management System
- Oracle Utilities Operational Device Management
- Oracle Utilities Work and Asset Management

Built on the Oracle Business Intelligence Enterprise Edition (OBIEE) platform with integrated spatial features, Oracle Utilities Analytics supports end-to-end analytic workflows including the ability to drill back to the source application data. The extractors and schema are designed with pre-built mapping between the source and the target. They support schema extensibility with built-in user-defined fields, dimensions, and measures.

Oracle Utilities Analytics data warehouse is a separate database from the operational database. The data warehousing involves large volumes of data used primarily for analysis. The data warehouse has the following features:

- Includes pre-built star schemas, pre-built metrics, pre-built reports, and pre-built dashboards.
- Makes data structures easily accessible for end user reporting needs.
- Retrieves large volumes of data quickly, which allows for fast rendering of the graphics that showcase Key Performance Indicators (KPIs).
- Contains star schemas and graphics suited for the data retrieved from the various Oracle Utilities edge applications.
- Provides you with the ability to add star schemas and graphics per your requirement using the required development tools.

Architecture

This section provides details on the product architecture and how the different components work together. The objective of this section is to provide you a high level understanding of the key components of the Oracle Utilities Analytics product and how these components work together.

- Product Overview
- Deployment

Product Overview

Selected tables from edge application databases are replicated to the corresponding replication schema using Oracle GoldenGate. The data in the replication area is transformed into appropriate star schemas using Oracle Data Integrator. Oracle Data Integrator is utilized to generate the aggregated materialized views based on the star schemas. Oracle Business Intelligence Enterprise Edition is used for presenting the data in the star schemas and the associated materialized views on dashboards for each subject area.

This diagram depicts the conceptual components involved in the Oracle Utilities Analytics product.
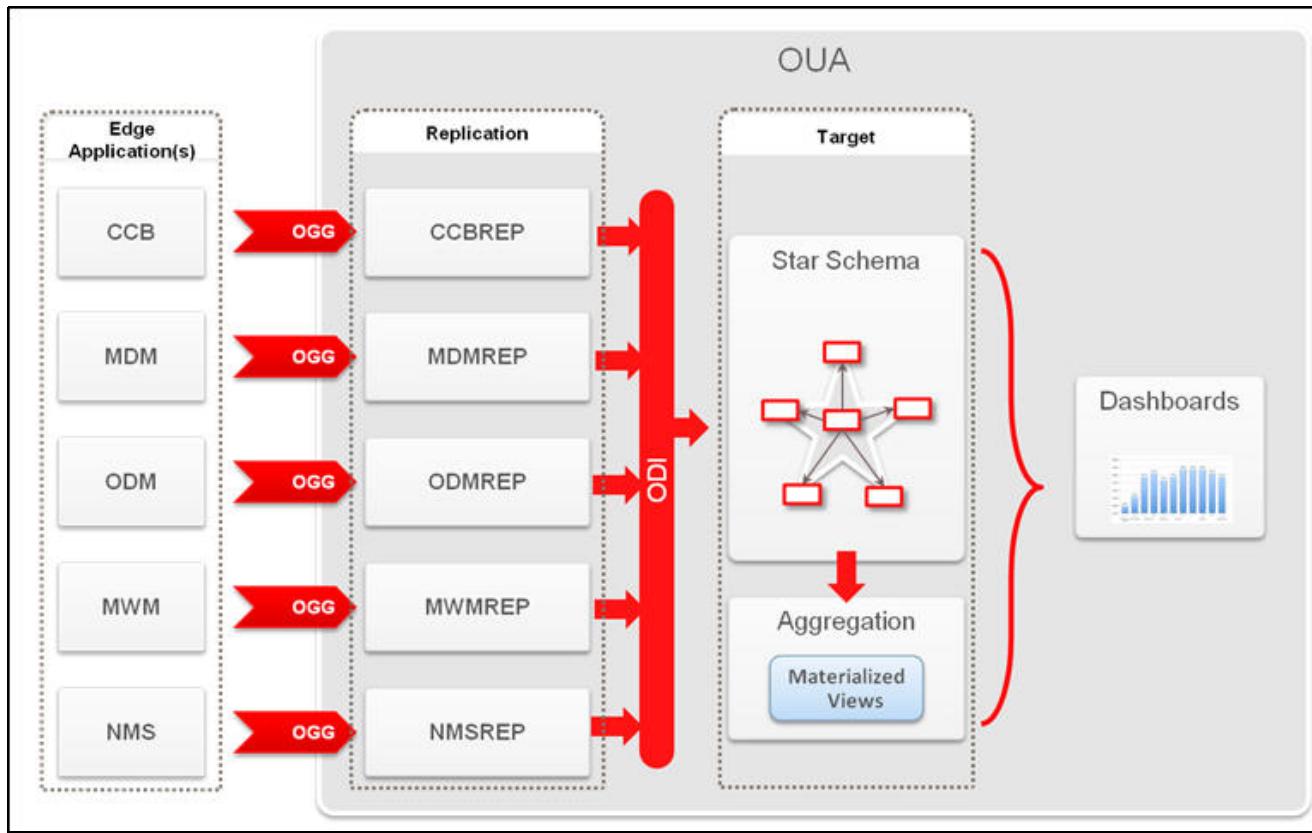
Deployment

A standard deployment of the Oracle Utilities Analytics product consists of a database and an application server. Two WebLogic installations are performed on the application server, one for Oracle Business Intelligence Enterprise Edition and another for Oracle Data Integrator. And, Oracle GoldenGate is installed on the source database server and on the target database server.

The diagram below depicts a standard deployment of the Oracle Utilities Analytics product.
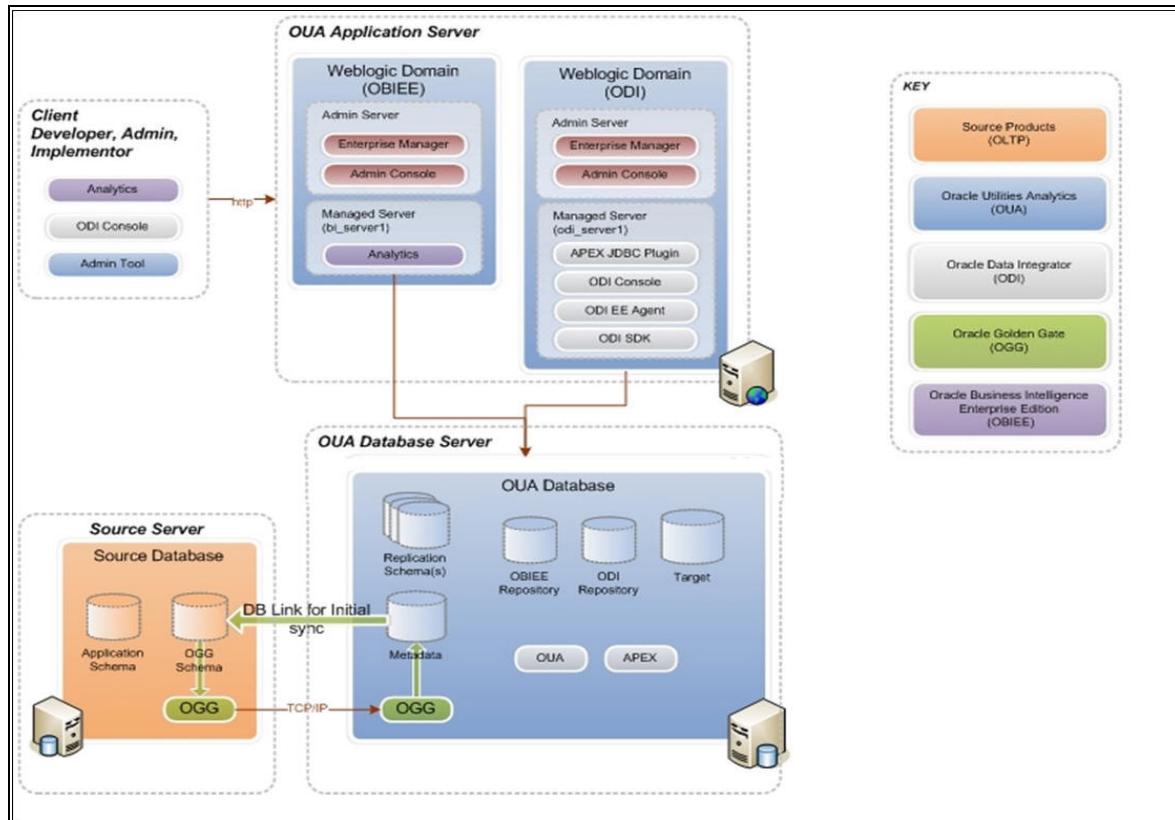
Chapter 2

Oracle Utilities Extractors and Schema

This chapter lists the various components related to the extractors and schemas within the product and how they work together in the product, including:

- Data Flow

- ELT Architecture

- Replication

- Default Keys in Dimensions

- Metadata Model

- Configuration Views

- Oracle Data Integrator Components

Data Flow

A data warehouse typically pulls data from multiple sources and transforms data stored in a relational model into a dimensional model. Oracle Utilities Analytics is a data warehouse built on data pulled from multiple Oracle Utilities products.

At the core of Oracle Utilities Analytics is the ELT (Extraction, Loading, and Transformation) processes, which extract data from source system tables and transform them into dimensional structure. These processes have been designed to be metadata-driven to ensure scalability and customizations without the need to change the out of the box components.

Data passes through four stages during the ELT process:

1. **Source**: This represents one of the source systems attached to Oracle Utilities Analytics. A predefined set of tables supports the load of the star schemas provided in Oracle Utilities Analytics.

    Note: Oracle Utilities Analytics supports multiple source products and up to nine instances of the same source product can be attached.

2. **Replicated:** Each instance of the source product gets a dedicated schema on the target database. This separates the objects from each source system and avoids any naming conflicts across multiple sources. The replicated tables are created without foreign key constraints to reduce overhead. The schema is created dynamically when a source product instance is attached to Oracle Utilities Analytics. The data is synchronized using Oracle GoldenGate.

3. **Staging**: Oracle Data Integrator processes use staging schema for storing data temporarily between transformations. The data in this schema is semi-persistent and is retained purely for debugging and traceability. The retention period is configurable.

4. **Target**: This is the schema, which stores the facts, dimensions, and materialized views used by the analytic dashboards.
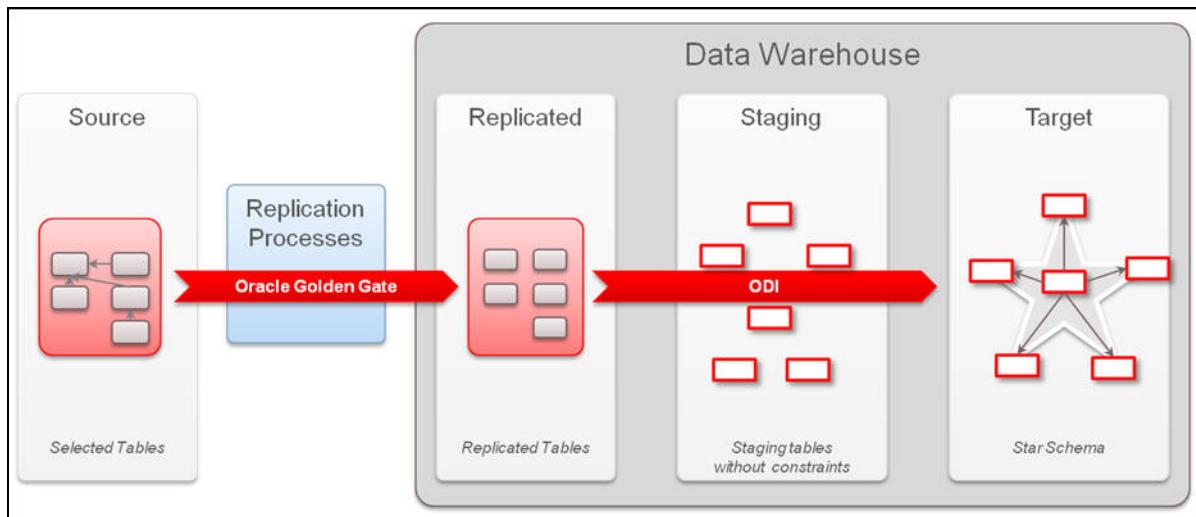
ELT Architecture

The diagram below shows the various components and how they are involved in the ELT architecture.
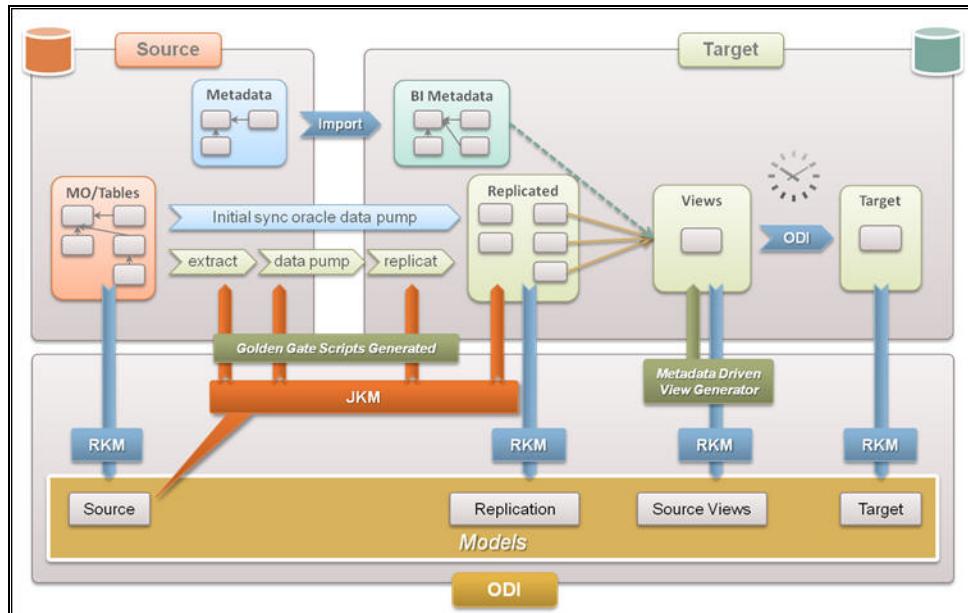
In this diagram, the ELT architecture has been divided into three sections.

- **Source**: This section contains source product instance components and components that are added to the source product environment to facilitate real-time synchronization.

- **Target:** This section covers components on the target Oracle Utilities Analytics database and application server to facilitate the initial and ongoing data loads into the star schemas.

- **Oracle Data Integrator**: This section contains the Oracle Data Integrator components that are leveraged for the configuration and the ongoing data loads.

After the installation, Oracle Utilities Analytics contains the metadata and the Oracle Data Integrator components designed to load data into the target star schemas using the replication schema objects. However, no source product instance is associated with Oracle Utilities Analytics. Attaching a source product instance is the first step after installation. This involves the following actions.

1. **Importing Source System Metadata**: Oracle Utilities Analytics imports source product metadata into its metadata structure. This involves creation of product instance entry and detecting some of the configurations that have been implemented on the source product.

2. **Creating Job Configuration Entries**: The job configurations with predefined defaults are created for all entities associated with the source instance. All jobs are disabled by default.

3. **Reverse Engineering Models**: This step creates one or more models (Oracle Data Integrator) and executes a custom Reverse-Engineering Knowledge Module (RKM) to import specific tables into each model. The association of tables to models is pre-configured. The tables are reverse-engineered at the time of implementation to ensure that any changes in the table structures are automatically captured.

4. **Enabling Ongoing Sync**: Once the source models have been created, the Journalizing Knowledge Module (JKM) is executed. This enables journalization on the tables of the model. The replication schema is created and the corresponding replica tables are created for each model. This step also generates "extract", "data pump", and "replicat" parameter files

for each model. These parameter files control the ongoing synchronization. The "extract" and "data pump" parameter files are deployed on the source and the "replicat" is deployed on the target.

5. **Performing Initial Sync**: Once these parameter files have been deployed successfully, the initial synchronization is performed. The data volume to be synced depends on how long the source product instance has been active at your location. The data sync is performed by executing an Oracle Data Integrator scenario, which internally utilizes Oracle Data Pump to copy data from the source database to the replica database.

6. **Generating Metadata Driven Views**: Once the replicated tables and the source instance specific metadata have been imported, the metadata-driven views are generated in the replication schema. These views are the sources for the Oracle Data Integrator interfaces that load the dimensions and facts.

7. **Enabling Jobs**: Having configured Oracle Utilities Analytics to work with the source instance to load the associated star schemas, the Oracle Utilities Analytics Administration user interface may be used to enable jobs, load the star schemas, and view the loaded data in the Dashboards.

Replication

This section describes about replication and DDL support including:

- Replication Patterns
- Structure of the Replicated Table
- DDL Support

Replication Patterns

The following patterns are utilized when replicating tables from the source. These patterns are based on how the tables are being utilized by the source system and which target entities are loaded from these tables.

Pattern	Description	Approach
Overwrite	The historical changes do not need to be tracked for these entities. These are typically the transactional or the master data used to populate the Type 1 dimensions.	• Any changes in the source are updated on the target replication table. • The type of transaction is marked on the replicated table. • The transaction timestamp is marked on the replicated table. • The rows deleted on the source are not marked as deleted on the replication layer.
History	Change history is not maintained in the source product, but the business intelligence tables are required to be tracked for populating Type 2 dimensions.	• Every change is treated as a new historical item. • The previous row for the same key must be marked with an end date. • The latest record is marked as current.

Pattern	Description	Approach
Effective-Dated	The source maintains history by using either an effective date column or effective start and end dates. If the end date is not defined, it is calculated for better responsiveness of the data load processes.	• A change in the source is updated as is. • The effective end date is calculated. • The latest record is marked as current.
Key Tables	Oracle Utilities Analytics Framework product stores the keys for the primary tables of a maintenance object. This data is not deleted when data from base tables are archived or purged. This is required to differentiate between a deletion on the source system versus an archival on the source system.	The table consists of key columns only and is replicated as is with the timestamp and the source system change number.
SCD2	The SCD2 behavior requires the usage of columns to store the effective start and the effective end date of a historical record. This may also include an optional column to identify the current row.	In the Oracle Utilities Network Management System source system, some of the tables follow the SCD2 behavior utilizing birth and death columns to store the effective start and effective end dates.

Structure of the Replicated Table

A replicated table has all columns of the source table and the following additional attributes depending on the categorization. The column order of the replicated table matches the column order of the source table and any additional columns added will be after.

The table below shows the different patterns of replication and the additional columns, which are present in each of the patterns.

Column	Overwrite	Effective Dated	SCD2	History	Key
All Source Columns	Yes	Yes	Yes	Yes	Yes
Slicing Timestamp	Yes	Yes	Yes	Yes	Yes
System Change Number	Yes	Yes	Yes	Yes	Yes
Last Change Timestamp	Yes	Yes	Yes	Yes	
Operation Indicator	Yes	Yes	Yes	Yes	
Cleansed Indicator	Yes	Yes	Yes	Yes	

Column	Overwrite	Effective Dated	SCD2	History	Key
Journal Effective Start Timestamp				Yes	
Journal Effective end Timestamp				Yes	
Effective End Date		Yes			
Current Indicator		Yes	Yes	Yes	

DDL Support

Any product can undergo data structure changes as a result of upgrades or customizations. The table below listssome possible edge application upgrade scenarios and the expected Oracle Utilities Analytics behavior. The DDL replication feature of Oracle GoldenGate is utilized to support structural changes.

Category	Change	Expected Behavior
Structure change	The table/column is renamed or dropped.	The DDL replication propagates the DDL change on the source to the target. • The renaming of a table is not supported in the DDL replication. However, rename of a column is supported in the DDL replication. • The dropping of a table is not supported, but dropping a column is handled by the DDL replication. • In case a table ior a column is renamed, views referring to the original tables and column are invalidated and target fact dimension loads fail. This requires manual steps to resolve the table/column rename.

Category	Change	Expected Behavior
Structure change	Column(s) added	The DDL replication adds the new column(s) before the JRN columns in the replication table and no components are impacted. Oracle GoldenGate processes continue to run and the new columns store values updated or inserted at the source. The restrictions are given below: • If a column is added to a HIST type table, then any subsequent updates on the added column are not captured if the update does not include a key column filter in the where clause. • The XMLTYPE columns are not supported as DBMS_REDEFINITION do not support online redefinition of tables with binary XML columns. • Every DDL replication that is applied on the replication layer via Oracle GoldenGate causes the replicat process to be stopped. This is required to prevent any data loss during the time that the replicated table is redefined. After every DDL replication, the replicat process needs to be restarted.
Structure change	Table(s) added.	No impact on Analytics.
Structure change	Column data type changed.	The corresponding change is applied on the replication layer and the new data is processed without interruptions.

Category	Change	Expected Behavior
Content change	CLOB xml structure change.	Assuming that this node: `<isResidentialCustomer>true </isResidentialCustomer>` is changed to: `<customerType>residential </customerType>` After the change: • None of the interfaces fails. • The logic to identify customer type as residential now returns null and all customers are now tagged as non-residential. • Data correction is required on the replication layer to sync the structures to match the new structure. This may require a change on the Analytics interfaces.
Content change	New/modified characteristic types	No impact on Analytics. The new characteristic types might need to be configured in the job parameters.

Category	Change	Expected Behavior
Content change	Data correction patch	After an upgrade where table structures have been changed, it is possible that a data correction patch is provided to convert the older version data to the newer version.
		• For the tables where you do not maintain history in the replication layer, this data correction may automatically propagate to the replication layer depending on the type of structural change.
		• For the tables where you maintain history, this causes a new row to be created on the target for each of the impacted tables. The older version data will still have the old data and require a cleanup on the replication layer to fix the data appropriately.
		• This potentially creates a spike in data volume to be reprocessed for the dimensions and the facts.
		• Depending on the functionality impact of the data correction, facts/dimensions may need to be reprocessed completely.
Version	Multiple instances on different versions.	Multiple instances of the same edge application product on the different versions is not supported.

Default Keys in Dimensions

All the dimensions have two default rows:

Primary Key	Default Values	Data Source Indicator	Remarks
0	***	NULL	This dimensional record is used as a reference for empty foreign key relation on a fact. If a particular fact record does not have any relation to a particular dimension, then the fact record is mapped to the 0th key in the dimension. For example, if no meter is installed at a service point in Oracle Utilities Meter Data Management, the service point fact record in the Service Point Fact points to the 0th key in the meter device dimension.

Primary Key	Default Values	Data Source Indicator	Remarks
-99	-N/A-	NULL	This dimensional record is used as a reference for missing foreign key relation on a fact. This scenario arises when a particular fact record does have reference to a particular dimension as per the source data, but the referenced value itself is not yet available in the dimension. In such cases, the fact record is mapped to the -99 key in the dimension.
			For example, consider that a meter has been installed at a service point in Oracle Utilities Meter Data Management, but a record for the meter has not yet been loaded in the meter device dimension. The service point fact record in the Service Point Fact points to the -99 key in the meter device dimension.
			This means that the ELT administrator should check for these -99 records in the fact and double check why the dimension record is not loaded. There can be two reasons:
			1. This can be a classic case of late arriving dimension, wherein the fact record is attempted to be loaded even before a referenced dimension record was not yet loaded. In such cases, the ELT framework automatically tries to map the fact record again. Whenever the dimension record gets loaded, the fact record would get automatically corrected.
			2. There can be other cases such as the source data reference for the dimension being incorrect or corrupt.

Referencing Foreign Keys

- The Foreign Key (FK) value is set to 0 for any rows with NULL values for "Ref Natural Key".

- The FK value is set to -99 for any rows with a value for "Ref Natural Key", but without a corresponding record in the dimension. The error reprocessing automatically binds these rows with the correct rows in the dimension once the dimension value has arrived.

Metadata Model

The objectives of the metadata model are:

- To maintain configuration details for each source system.

- To maintain configuration of each instance of the source system when multiple instances of the same source system have been implemented.

- To maintain configuration for each of the target entities, reusing interfaces and packages for multiple instances.

- To manage dependencies.

- To manage multiple Data Source Indicators (DSIs) allowing external references at the instance level and also at granularity of individual rows of the source data.

You should not change any of the metadata entries that are delivered out of box in these metadata tables. Wherever your inputs are needed/allowed, the Oracle Utilities Analytics Administration Tool supplied with the Oracle Utilities Analytics product should be used to provide inputs. Other than this method, any other means to change the metadata is strictly prohibited.

This section describes the metadata model utilized by the implementation, including:

- Products

- Instances

- Global Configuration

- Entities

- Job Configuration

- Schedule

- Objects

- Mapped Objects

- Source Tables

- Job Executions

- Dependencies

- DSI Mapping

- Server Configuration

- GoldenGate Checkpoint

- Extract Parameters

- Storage Configuration

Products

This table is utilized to store a list of products that are supported by the implementations. This table is used to identify which products are supported and to identify the Oracle Data Integrator scenario to be used to import source metadata and configurations.

Instances

You may have multiple instances of the same product that you want to integrate with Oracle Utilities Analytics. The instance object represents each instance of the same product that can be utilized as a source. The objective is to enable development of a single interface that can be utilized across multiple instances.

The instances differ by the source database connections and possibly by differing configurations. The Data Source Indicator (DSI) is unique across instances and products as this provides the ability to identify the source of the data in the data warehouse.

Global Configuration

This table is utilized to store the configuration name value pairs that are applicable on the product. These configurations are utilized to control the execution modes for interfaces.

Entities

This table holds the configurations for the target entities in the data warehouse. The target entities can be dimensions, facts, or materialized views. The same target entity may be loaded from multiple source instances. The table stores the configuration that is common across multiple instances.

Job Configuration

This table is used to provide configuration for the package (the executable logic) for populating the target entities for each instance. This table stores the current state information and an override package for use when the logic for one instance differs from the logic used for other instances. The override package name can be used to provide a custom scenario in case you want to deviate from the logic provided.

Schedule

This table is used to provide configuration for scheduling jobs at regular intervals. This table also stores the start and end time of the job within a day to enable the job to run during a specific time period.

Objects

This table is utilized to store all objects and configurations that are common across all objects. The objects can be the source tables that need to be replicated, the database objects that need to be generated, or source Maintenance Objects (MOs) or Business Objects (BOs). Any object that is created during installation or when a source is attached is listed in this table along with configuration associated with the object.

Mapped Objects

This table contains a list of all source objects mapped to a target objects. This is the primary data entry point mapping a source MO/Table with the target entity. This is used to map views to be generated for the target entities. The entries are identified based on object type flag. Below are the possible values and their usage.

Object Type flag	Description
MO	**Maintenance Object**: The value in the source object field is identified as OUAF Maintenance object based on this flag.
TBL	**Table**: The value in the source object field is identified as an NMS source table based on this flag
PRVW	**Primary View**: The value in the source object field is identified as primary view to load the data into the target object based on this flag
MVDP	**Materialized View Dependency**: Based on this flag, it is assumed that the value in the source object column is a dimension or fact entity and the value in the target object is a materialized view. This also indicates that the materialized view load should depend on the load of the entity specified in the source object column.

Object Type flag	Description
DMDP	**Dimension Dependency**: Based on this flag, it is assumed that the value in the source object column is a dimension entity and the value in the target object is a fact entity. This also indicates that the fact load should depend on the dimension load.

Source Tables

This table contains configurations controlling which tables are replicated and the mode of replication.

Job Executions

This table is used for tracking the execution of the ELT processes. An entry is created for each execution. Some attributes are populated from the SNP_SESSION table which is used by Oracle Data Integrator to track sessions.

Dependencies

This table is used to map a job to the jobs it is dependent on. The dependent objects are only executed up to the minimum sync timestamp of all of the dependencies.

DSI Mapping

A source system may be integrated with other source systems. When pulling information from multiple source instances, it is possible that the data in one instance refers to the master data from another instance. This table allows such cross references to be configurable. By default, the configuration is at an instance level that is a product and instance number level. However, more fine-grained control is possible by specifying the entity for which the configuration should be applied.

Server Configuration

This table is used for generating the Oracle GoldenGate scripts for ongoing sync. This stores database, GoldenGate, and JAgent specific information of the target and source system.

GoldenGate Checkpoint

This table is used by Oracle GoldenGate replication process to track its processing activities. This table is used by the scheduler process to ensure that the warehouse loading tasks only process data that has been synced. The structure is controlled by Oracle GoldenGate.

Extract Parameters

This table stores configurations that control the extraction of data. Each source system may have different codes used for the same purpose. This table allows you to configure the codes so that the extraction routine can utilize it appropriately.

Storage Configuration

This table is utilized to store configuration for allocation of table spaces while creating objects within the knowledge models. This table has the default configurations at the pre-loaded "Global" context level.

Configuration Views

This section covers the following views:

- Context Configuration
- Replication Configuration
- Job Configuration
- Storage Configuration
- Dependencies
- Schedule
- Jobs
- Ready for Load
- Wait Reasons

Context Configuration

The view associates the product instance with the logical and physical schemas defined in Oracle Data Integrator.

Replication Configuration

The replication configuration view is a summary of the settings that are used to control the tables to be replicated.

Job Configuration

The job configuration view shows all the jobs based on the configuration in the metadata table. The schedule date and the slice start and end date are calculated along with the maximum retry interval, retry interval, and Staging retention period.

Storage Configuration

The storage configuration view shows the tablespaces configured for different object types for each logical schema. The configuration can be common or can be granular at logical schema level. This view breaks down the configurations at the lowest granularity and provides easy to access all attributes of storage configuration handling the complexities of identifying context and logical schema specific configurations that override the global configurations.

All the attributes of storage configuration handling the complexities of identifying context and logical schema specific configurations that override the global configurations.

Dependencies

The Oracle GoldenGate processes can be tracked via the Oracle GoldenGate Checkpoint table. A source instance can have multiple Oracle GoldenGate processes and each process consists of multiple tables. A target entity can be loaded only up to the maximum sync timestamp of all the Oracle GoldenGate processes associated with the context in which it is executing.

A fact load is dependent on the load of Type 2 dimensions associated with the fact. The fact associated with a Type 2 dimension cannot be loaded beyond the time that the dimension has been synced.

A materialized view becomes stale if any of the underlying tables is modified. Hence, the materialized view depends on the execution of any of its underlying tables.

The view identifies the base object and the timestamp associated with its dependencies so that appropriate actions can be taken to avoid any data loss.

Schedule

This is a consolidated list of the start time of the job for a given schedule. Based on the start time at regular interval the jobs are picked for execution.

Jobs

This is a consolidated way of viewing all jobs that are being executed or have completed. The view shows the slicing units, duration of execution, and volumes being processed. This can be utilized for health checks and for monitoring the load processes.

Ready for Load

This view governs the selection of jobs for execution and controlling the pattern of execution. This is governed by the following rules.

- The entity should not be disabled.

- The configured scenario should exist.

- All dependencies should have run.

- The number of running/error instances of the job should be less than the maximum parallel executions allowed.

- If the instance has been configured as run once, then it should not execute once it has been successfully executed.

- If a job fails, it should be retried until the maximum retries per day is reached. The interval between successive retries should be based on the configuration.

- The Oracle GoldenGate models comprising of the source tables used in the entity should have been synced. In case the sync timestamps vary across multiple models, then the least common sync timestamp is used.

- The snapshot entities are executed on or after the snapshot end period.

- The schedule time can be used to stagger loads and distribute processing. A job is not executed until the current time crosses the scheduled time.

Wait Reasons

This view can be used to diagnose the reasons why a job is not being executed. This view applies the rules for scheduling and reverses the logic to obtain the reasons for not executing. The following issues can be identified using this view.An entity is deactivated.

- The Oracle GoldenGate replication is not configured for model {model name}.

- The configured scenario does not exist.

- This job is scheduled to run at {timestamp}.

- This one time load interface has already executed successfully.

- Reached max retries allowed per day.

- The approximately {X} minutes until next retry.

- The maximum number of parallel materialized views execution reached.

- Waiting for a change to occur in dependent entity.

- Waiting on a dependent entity to be loaded.

- Waiting for at least one minute of incremental data to arrive.

- The cut-off time has been enabled. All the entities are loaded only up to {timestamp}.

- The temporary table is not reversed in the staging model.

- The staging table is not reversed in the staging model.

- The flex fields are not set for table.

Oracle Data Integrator Components

This section lists the high level components in Oracle Data Integrator:

- Project

- Model Folders

- Scenario Folders

- Oracle GoldenGate Logical Schemas

- Connections

- Contexts

Project

The out-of-the-box implementation installs the "Oracle Utilities BI" project. This project contains all the out-of-the-box code for loading facts, dimensions, and materialized views. This project and all its underlying objects may be modified in subsequent releases.

All the customizations should be done in a separate project.

> **Note**: Any custom objects if created within this project may be deleted during a subsequent release.

Model Folders

Oracle Utilities Analytics comes with a set of predefined model folders and models. Each product model folder consists of the below model folder and model.

Name	Purpose
Source	The model folder is created when the first source product instance is attached to Oracle Utilities Analytics. This folder contains all models associated with a source instance. These models are also created when a source instance is attached to Oracle Utilities Analytics.
Replication	This folder contains models for all the replicated table objects and the views generated in the replicated schema.
Dimensions	This folder contains the models for the star schema dimensions. The shared dimensions, which are populated from multiple edge applications, are grouped together.
Facts	This folder contains the models for the star schema facts.
Staging	This folder contains the intermittent tables created for loading the Dimension/Facts.

Name	Purpose
User Defined	This folder contains the user defined tables that are utilized for customization of dimension and facts.
Metadata	This contains the models for all metadata tables.

The objects within these models should not be modified. All custom models should be created under a separate custom model folder.

> **Note**: Any custom objects created within the Oracle Utilities Analytics models may be deleted during a subsequent release or upgrade.

Scenario Folders

All scenarios that are delivered are grouped into scenario folders based on the functionality and usage for a product. The Materialized Views, Replication, Dimension, and Facts folder are organized under each product configured with Oracle Utilities Analytics. The following scenario folders are available.

Name	Purpose
Framework	All of the framework components are included in this folder.
Accelerators	Implementation accelerators for each source product framework are available in this folder. These scenarios are executed automatically during the steps for attaching a source instance.
Materialized Views	All the scenarios that are used to create or refresh materialized views are organized in this folder. These are automatically executed based on the configured schedule.
Replication	All scenarios that are used to create views in the replication schemas are organized in this folder. These are executed when attaching a new source instance to Oracle Utilities Analytics or when the view generator process is executed.
Dimensions	All the scenarios that are used to load the dimension of star schema tables are organized in this folder. These are automatically executed based on the configured schedule.
Facts	All the scenarios that are used to load the fact tables in start schema are organized in this folder. These are automatically executed based on the configured schedule.

> **Note**: Any custom objects if created within these folders may be deleted during a subsequent release.

Oracle GoldenGate Logical Schemas

The logical connections as defined in Oracle Data Integrator are:

Logical Name	Purpose
Source	The logical connection associated with the source data. This contains all the tables that will be replicated.
Journal	The journal schema is utilized as the work schema where objects can be created for processing. This is also the user connecting to the source system and managing the GoldenGate replication.

Logical Name	Purpose
Replication	The logical connection associated with the replicated area where the data is replicated and history is maintained before the warehouse load process is initiated.
Staging	The logical connection associated with the staging area where the semi-persistent staging tables reside.
Target	The logical connection for the final target star schemas.
Repository	The logical connection to the work repository. This is used for internal objects associated with the Oracle Utilities Analytics processes.
Master	The Oracle Data Integrator master repository is maintained in a separate schema associated with the master logical schema.
Metadata	The metadata configurations utilized to manage the solution are stored in this data area.
Configuration	This is a flat file logical schema used for import and export of the primary metadata.

Connections

During installation a single connection named "B1" is created to the target database. This connection is used for all data load processes and it connects using the metadata user. Subsequently when attaching a source product instance, the additional connections are created. These connections are named based on the source product code and instance number.

Contexts

The default context "Global" is the only context created during install and this is automatically mapped to the appropriate logical schemas during installation. Additional contexts are created when adding source instances to Oracle Utilities Analytics. The contexts map the physical schemas to the logical schemas based on the source connection and the mappings defined in the "Global" context for the target schemas. The jobs for a particular source are always executed in the context defined for that source instance.

Note: The contextual mapping should not be modified manually.

Chapter 3

Oracle Utilities Analytics Dashboards

A number of pre-built analytics are delivered with Oracle Utilities Analytics. They cater to a rich set of KPIs for each source product whose data can be extracted into the data warehouse. The Dashboard components in Oracle Utilities Analytics are described in the following sections:

- Dashboard Catalogs
- Oracle Business Intelligence Repository

Dashboard Catalogs

Oracle Utilities Analytics. delivers multiple dashboards for each source product. The dashboards are grouped in logical divisions and are a part of a Presentation Catalog. The table below provides the mapping between the various source products and the catalogs that are available for them.

Source Product	Presentation Catalog in Oracle Utilities Analytics
Oracle Utilities Customer Care & Billing	Credit & Collection Analytics Customer Analytics Revenue Analytics Exception Analytics
Oracle Utilities Meter Data Management	Meter Data Analytics
Oracle Utilities Mobile Workforce Management	Mobile Workforce Analytics
Oracle Utilities Network Management System	Outage Analytics Distribution Analytics
Oracle Utilities Operational Device Management	Operational Device Analytics
Oracle Utilities Work and Asset Management	Work & Asset Analytics

The details about each dashboard are described in detail in the *Metric Reference Guide* for each of the analytics above. The details about the star schema tables underneath the dashboards are covered in the *Data Mapping Guide* for each source product. All dashboards delivered with the base product are supported via a dedicated materialized view layer. These materialized views ensure that data is pre-aggregated and pre-joined as per the report requirements so that data is instantly available for reports. Data in the materialized views is refreshed via separate ELT jobs that get triggered after an ELT job loads data in a star schema table.

A special Administration catalog is provided along with the source products dashboards. The dashboards in the Administration catalog cater to the administration of the labels to be used in report and some map related configuration. These dashboards are covered in detail in the Configuring the Analytics section of Chapter 5: Configuring Oracle Utilities Analytics. You can add dashboards and reports to this catalog or choose to customize any of the existing ones. The details are described in the *Oracle Utilities Analytics Developer's Guide*.

Oracle Business Intelligence Repository

Just like in any other Oracle Business Intelligence Repository file, you can find all three layers in the repository file for Oracle Utilities Analytics. This section describes what is available in each of these layers in Oracle Utilities Analytics.

- Physical Layer
- Business Model and Mapping Layer
- Presentation Layer

Physical Layer

The physical layer in the Oracle Utilities Analytics repository file has the following physical data construct groups:

- **Initialization Blocks:** This data construct is used as the source for the repository initialization block queries, which are used to initialize the session variables that are used in

various dashboards. There are two connection pools defined; one for the data warehouse and the other for the Oracle Utilities Meter Data Management system source database, which hosts the aggregator materialized views for some of the dashboards in Meter Data Analytics

- **Materialized Views:** This data construct is used as the source for fetching the materialized view data from the data warehouse.

- **MDM Database:** This data construct is used to connect to the Oracle Utilities Meter Data Management system source database to fetch the data from the aggregator materialized views for some of the dashboards in Meter Data Analytics

- **ODIDEV:** This data construct defines a special connection pool with write access to certain sets of tables in the data warehouse. These are tables used in the Administration group of dashboards where data warehouse data is updated using the Oracle Business Intelligence Enterprise Edition's writeback feature.

- **Oracle Data Warehouse:** This data construct defines the connection pool to fetch the data from the star schema tables from the data warehouse.

Business Model and Mapping Layer

The BMM layer in the repository file for Oracle Utilities Analytics has the following groups:

- **Core:** This group defines the data model of all star schemas in the data warehouse. The dimensional hierarchies are also defined here. Any logical columns needed for calculations on top of fact measures or dimension attributes are defined here. The materialized view mapping is also done here. Each fact and dimension has the logical mapping to one or more materialized views, which were designed for it. This ensures that the reports based on the respective fact or dimension goes to the materialized view first in the same order as the logical sources are defined

- **OUA:** This group defines the data model of the special set of tables where writeback capabilities are enabled.

Presentation Layer

The Presentation Layer subject areas are defined per the star schema in the data warehouse. In most of the subject areas, there is one fact along with all of its associated dimensions. In some cases, multiple facts might be present to support certain complex report requirements. Each subject area name starts with a three letter prefix of the source product it is based upon. This is the layer that inserts the label mechanism in the product, which provides the Oracle Utilities Analytics dashboards with multi-language support. Each presentation layer field is associated with a session variable name. The session variables are dynamically mapped to label descriptions of an appropriate language.

Chapter 4

Oracle Utilities Analytics Administration Tool

The Oracle Data Integrator-based Extract, Load, and Transform (ELT) architecture is a metadata driven framework that allows you to configure out-of-the-box ELT jobs and makes it easy to integrate custom ELT jobs to Oracle Utilities Analytics. The out-of-the-box ELT jobs are set up during installation; however, you can maintain some of the parameters using Oracle Utilities Analytics Administration Tool. This tool is based on the Oracle Application Express (APEX) platform. The following tasks can be performed using Oracle Utilities Analytics Administration Tool:

- Configuring Global Settings

- Configuring Server

- Configuring Product Instance

- Configuring Storage

- Configuring Source Table

- Configuring Target Entities

- Configuring Schedule

- Configuring Jobs

- Customizing the Column Properties of Type 2 Slowly Changing Dimension

- ETL Replication View

- NMS Configuration

- ETL Job Execution

Configuring Global Settings

This table is used to capture the global settings needed for Oracle Utilities Analytics:

Attribute	Purpose
Product	A reference to the product code that identifies the product for this instance. This could be one of the configured source products or the base Oracle Utilities Analytics product itself.
Instance Number	The unique number of the configured source product instance. For the base Oracle Utilities Analytics product configuration settings, this value is set to null.
Description	This column describes the purpose of the specific global configuration setting.
Value	The configuration value that you can update based on their setup.
Data Type	The data type of the configuration value that a user is expected to enter.
Data Format	The specific format of the data that a user is expected to enter the configuration value in.

The **Global Configuration** page in the Administration Tool allows you to enter a configuration value for each configuration setting. As a part of the Oracle Utilities Analytics installation process, the global configuration records are generated for the configured source application and the target Oracle Utilities Analytics application. You can then use this page to change the configurations.

You are expected to enter the configuration value strictly in the specified data format only. Entering value in other formats results in errors during the ELT processing.

Global Configuration

Product	Instance Number	Description	Value
Customer Care and Billing	1	Product Extract Date - The date from which all ETL jobs for the specific product instance will be configured to start the initial load	01-Jan-2001
B1	(None)	Number of days for which the logs would be retained	11
B1	(None)	Maximum number of MV interfaces which can run in parallel	6
B1	(None)	Database Edition Type - Mention the type of the Oracle Database, whether it is Standard Edition(SE) or Enterprise Edition(EE)	EE
B1	(None)	Enable Partitioning - If a partitioning liscense has been purchased from Oracle, set this to Y	N
B1	(None)	Cut Off Time - The time of the day upto which all changes from the source will be loaded by the ETL jobs. Any changes after this will not be loaded on the same day. Set this to default value (i.e. 23:59:59) to keep loading data continously.	23:59:59
B1	(None)	Global Extract Date - The date from which all ETL jobs will be configured to start the initial load	01-Jan-2001

Configuring Server

Oracle Utilities Analytics uses Oracle GoldenGate to capture changed data. This table is used by the Oracle Data Integrator to generate the GoldenGate extract and replication process and deploy the same in source and target servers using the JAgent configuration.

Attribute	Purpose
Context	The code is either the context code identifying the instance uniquely or B1 for the target data warehouse.
Host Name	The host name of the data server based on the server type.
Port	The port that the data server utilizes.
DB Service Name	The service name of the database.
Home Path	The home folder of the data server as defined by the server type.
Username	The username to connect to the data server. For example: • The database user if the server type is database. • The server user that has access to the software if the server type is GoldenGate or JAgent.
DB Client Home	The database client home path for the GoldenGate server.
Server Type	The type of server to identify the different configurations. For example: • Data Base: Details about the database. • Golden Gate: Details about the GoldenGate • JAgent: Details about the JAgent configuration

The **Server Configuration** page in the Administration Tool allows you to view the server details. As part of the Oracle Utilities Analytics installation process, Oracle GoldenGate, Oracle Database, and JAgent configuration records are generated for the configured source application and the target Oracle Utilities Analytics application.

Configuring Product Instance

The Product Instance represents a specific instance of a source application that can be configured as a source for Oracle Utilities Analytics solution. A record is created for every product instance where ELT is setup to extract data. Oracle Utilities Analytics delivers a record for the product that uses Oracle Data Integrator for the ELT.

> **Note:** You may have multiple instances of the same source application to integrate with the Oracle Utilities Analytics data warehouse.

The following attributes are available on the Product Instance metadata table.

Attribute	Purpose
Product Code	A reference to the product code that identifies the product for this instance.
Instance Number	A unique number starting with 1 to uniquely identify the instance.

Attribute	Purpose
Context Code	A unique code comprised of the product code and instance number that is used to identify the connections in Oracle Data Integrator. Due to limitations imposed by Oracle GoldenGate (used for replication), the context code cannot be more than 5 characters in length.
Change Data Capture	A flag identifying the methodology to be used for replicating source tables. The default is Oracle GoldenGate.
Drill Back URL	A URL to be used to allow you to drill back to the source system from the analytics.
Time Zone Code	The time zone of the source product instance. This allows the interfaces to be built so that dependencies on job execution for multiple time zones are handled correctly.
Currency Code	The currency used in the source product instance. Multiple currencies are not supported.
Language Code	The primary language supported by the data warehouse. This is used to filter language specific data in the data warehouse.
Data Source Indicator	A unique value representing the instance. For Oracle Utilities Application Framework products, this is the environment ID of the source instance.
Source Instance Version	The current version number of the source product instance. This can be used to validate whether the version is supported by the data warehouse or provide suggestions if upgrades or patches need to be applied to enable support for the source product.

The **Product Instance** page in the Administration Tool allows you to maintain the existing records. To add a new product instance, use the script delivered as a part of the installation package

For more details, refer to the **Configuring the Source** sub-section under the **Upgrading Oracle Utilities Analytics Based ELT Component** section in the **Chapter 5: Upgrading Oracle Utilities Analytics** of *Oracle Utilities Analytics Installation Guide.*

Configuring Storage

An additional storage configuration is provided that enables you to manage the tablespaces utilized for the database objects that are created during the execution of Oracle Data Integrator jobs.

The following attributes are available:

Attribute	Purpose
Context Code	Identifies the context for which this is applicable. A value of "GLOBAL" is allowable across all the contexts unless overridden by an entry for the specific context.
Logical Schema	The logical schema for which the configuration is applicable. A empty value indicates that this is applicable to all logical schemas.
Object Type	The type of object for which the configuration is applicable. This can be Table, Index, or Partition.
Sequence	The default is 0. This is applicable only for partitions. This can be used to map multiple tablespaces in round-robin scheduling while creating the new partitions within the same table.
Tablespace Name	The name of the tablespace to be associated with the object.

The **Storage Configuration** page in the Administration Tool allows you to maintain the existing records related to configuration. Records are delivered out of the box for the star schemas tables delivered with Oracle Utilities Analytics.You cannot delete base product owned records.

Create new records for any entity that you want to add to the ELT. For owned records, you have full access to edit and delete them.

Configuring Source Table

This table contains configurations controlling, such as the source tables to be replicated and the mode of replication.

Attribute	Purpose
Source Product	A reference to the product code that identifies the product for this instance.
Table Name	The name of the source table. There is also a corresponding entry in the objects table.
History Type	The following scenarios exist: • Source system tracks history using an effective date column. • No history in source, but history needed for the Type 2 dimensions. • No history required.
Effective Date Column Name	The column name used for storing the effective dates in the source.
Characteristic Entity	Identifies the char entity flag in source Oracle Utilities Application Framework application.
Base Replication	Controls whether the table is required to be replicated for target entity load.

Attribute	Purpose
Group Number	Defines the grouping of various tables for a source product. The group numbers are used to define the various GoldenGate models that are set up to extract data from the source system.
Custom Replication	Extension for customizations. The additional tables to be marked for replication.
Purge Enabled	Controls whether the replicated table should be purged or not.
Replication Retention Days	The number of days the data should be retained in the replication layer.
Owner Flag	Indicates whether the record is owned by the base product (B1) or you (CM).

The **Source Table** page in the Administration Tool allows you to add, edit, and delete records. For this table, records are delivered for the source tables that are configured to be part of the out-of-the-box ELT. For these records, you can edit only the selected fields. You cannot delete base product owned records. You have full access to edit and delete owned records.

Configuring Target Entities

The table holds the configurations for target entities and their attributes in the data warehouse. The target entities can be dimensions or facts. The same target entity may be loaded from multiple source instances. Configuration that is common across multiple instances is stored in this table.

Attribute	Purpose
Entity Name	The name of the entity that needs to be scheduled for loading into the target.

Attribute	Purpose
Entity Type	The type of entity supported by Oracle Utilities Analytics: • Slowly changing dimension Type 1 • Slowly changing dimension Type 2 • Accumulation facts • Snapshot facts • Materialized views
Maximum Parallel Executions	It may be necessary to execute multiple instances to efficiently load data with each instance working on a different data set. This attribute controls how many parallel executions can be spawned for a single entity load.
Maximum Retries	This attribute controls the maximum number of retry attempts in a day. If the load is still failing when this limit is reached, it is retried next day.
Retry Interval	The base architecture has been designed for automatic retries. In case of failures, jobs are retried and this attribute controls the interval between successive retries. Default is 30 minutes, but it can be configured according to your requirements.
Scheduling Type	Three different modes of schedules are supported: • **Daily Incremental Load**: Loads are executed as soon as data is available to load. • **Near Real Time Load**: Loads are executed within a configured interval. This is for future use. • **One Time Load**: A load is executed only once.
Slice Duration Type	A slice is a volume of data bound within a time span (duration). Different objects have differing data distribution and load processing requirements. This attribute controls the duration between two slices. The following slicing intervals are supported: • Day(s) • Week(s) • Month(s) • Quarter(s) • Year(s) • Hour(s) • Minute(s)
Slice Duration	The number identifying the slice duration based on the slice duration type.
ODI Package Name	The name of the Oracle Data Integrator scenario that should be executed.
Staging Retention Days	The number of days to retain data in the staging tables.

Attribute	Purpose
Owner Flag	Indicates whether the record is owned by the base product (B1) or you (CM).

The **Target Entity** page in the Administration Tool allows you to add, edit, and delete records. Records are delivered out-of-the-box for the star schemas tables delivered with Oracle Utilities Analytics. For these records, you can edit only select fields. You cannot delete base product owned records. Create new records for any entity that you want to add to the ELT. You have full access to edit and delete owned records.

	Target Entity Id	Entity Name	Entity Type	ODI Package Name	Staging Retention Days	Owner
✎	119	CD_BILL_CAN_RSN	Dimension SCD1	B1_PKG_CD_BILL_CAN_RSN	7	B1
✎	118	CD_ASSET_INSTORE_AGE	Dimension SCD1	B1_PKG_CD_ASSET_INSTORE_AGE	7	B1
✎	117	CD_ASSET_INSTALL_AGE	Dimension SCD1	B1_PKG_CD_ASSET_INSTALL_AGE	7	B1
✎	116	CD_ASSET_INSP_STATUS	Dimension SCD1	B1_PKG_CD_ASSET_INSP_STATUS	7	B1
✎	115	CD_ASSET_DISP	Dimension SCD1	B1_PKG_CD_ASSET_DISP	7	B1
✎	114	CD_ASSET_AGE	Dimension SCD1	B1_PKG_CD_ASSET_AGE	7	B1
✎	113	CD_APPT_TM_OF_DAY	Dimension SCD1	B1_PKG_CD_APPT_TM_OF_DAY	7	B1
✎	112	CD_APPT_TM	Dimension SCD1	B1_PKG_CD_APPT_TM	7	B1
✎	111	CD_ALARM_LOG	Dimension SCD1	B1_PKG_CD_ALARM_LOG	7	B1
✎	108	CD_ACCT_TYPE	Dimension SCD1	B1_PKG_CD_ACCT_TYPE	7	B1
✎	252	CD_W_PLANNER	Dimension SCD1	B1_PKG_CD_W_PLANNER	7	B1
✎	251	CD_W_ASSET_INSTALL_AGE	Dimension SCD1	B1_PKG_CD_W_ASSET_INSTALL_AGE	7	B1
✎	250	CD_W_ASSET_AGE	Dimension SCD1	B1_PKG_CD_W_ASSET_AGE	7	B1
✎	249	CD_WO_STATUS	Dimension SCD1	B1_PKG_CD_WO_STATUS	7	B1
✎	248	CD_WO_ACTIVITY_STATUS	Dimension SCD1	B1_PKG_CD_WO_ACTIVITY_STATUS	7	B1

Configuring Schedule

The **Schedule Configuration** page in Oracle Utilities Analytics Administration is used to configure a job's scheduled start and interval. Multiple schedule types can be created with different schedule periods and jobs can be assigned to these schedules; the various job schedule may be distributed so multiple jobs do not run simultaneously. The Schedule period has to be created first and then the same has to be updated in the **Job Configuration** page (see **Configuring Jobs** on page 4-10).

Attribute	Purpose
Schedule Type	There are four schedule types: • **Incremental Update:** Load the data when the data is changed in the source. • **Near Real Time Load:** Loads are executed within a configured interval. • **Daily Refresh:** Load the data once daily (full load). • **One Time Load:** A load is executed only once.
Schedule Start Time	The time in a day when the load can be started.
Schedule Interval	The time interval between the consecutive execution of jobs.
Schedule End Time	The time in a day after which the load cannot be started.

Configuring Jobs

The **Job Configuration** page in the Administration Tool is used to provide configuration for the package (*i.e.*, the executable logic) for populating the target entities for each instance.

Attribute	Purpose
Source Product	A reference to the product code, which identifies the product for this instance.
Instance Number	To be populated for instance specific deviations. Ideally most data should be reusable across instances of the same product.
Target Entity	A reference to the entity.
Entity Active Flag	Indicates whether the ELT job should be picked up for execution by the Oracle Data Integrator Scheduler.
Slice Start Date/Time	The starting timestamp for the next execution.
Initialize Flag	Controls the initialization of the entity for slicing.
Execution Sequence	Sequence of execution within the same group.
Last sync timestamp	Stores the timestamp of the most recent successful slice.
Override ODI Package Name	Overrides the package name defined at entity level. This is to be used if the interface implementation for a specific instance differs from the common interface.
Override Slice Duration Type/Override Slice Duration	The override slice duration fields act as the overriding value for the ones in target entity. • **Type:** The override duration unit of measure (*e.g.*, **Year**). • **Duration:** The value for the override slice duration.
User Exit Procedure	The procedure name, which can be used to load the user-defined columns in the entity.

The **Configuring Jobs** page in the Administration Tool allows you to add, edit, and delete records. Records are delivered out-of-the-box for the star schemas tables delivered with Oracle Utilities Analytics. You cannot delete base product owned records. Create new records for any entity that you want to add to the ELT. You have full access to edit and delete owned records.

Maintaining Job Configuration

You can use the **Configuring Jobs** page to change job configurations. If you click the **Add** button or the **Edit** button ( ) for a selected job record row, the **Maintain Job Configuration** page opens allowing you to specify the job configuration.

Enabling Jobs

Jobs can be enabled from the **Jobs Configuration** page. Each job can be individually enabled by using the **Edit** button. Alternately, jobs of a group can be enabled together using the **Enable Jobs** button. There are three options for the job type:

- **Dimension Jobs**

- **Facts**

- **Facts and Materialized Views**

When the selected the **Job Type** is either **Dimension Jobs** or **Dimension and Fact Jobs**, the replication objects to be loaded with Dimensions and Facts are also enabled.
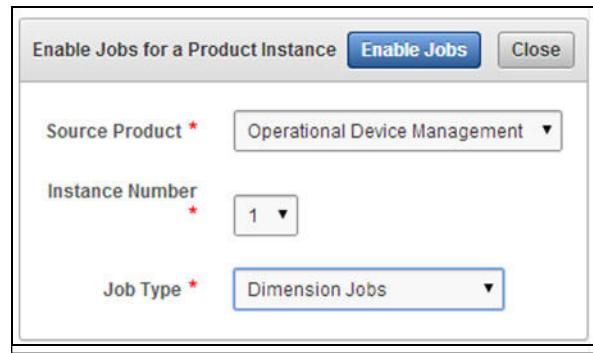
Customizing the Column Properties of Type 2 Slowly Changing Dimension

A Type 2 slowly changing dimension has the following column types:

- Natural Key

- SCD Start

- SCD End

- SCD Flag

Any change in any of the remaining columns is treated as a new history item for the same natural key. In the next release of Oracle Utilities Analytics, you will be able to decide which of the columns should not be considered for creating a new history item. Using Oracle Data Integrator, this is done by changing the corresponding property of the column in the model of the target dimension. However, this change requires regeneration of the dimension load scenario. The customizations that you do will get overridden by any future upgrades provided by Oracle.

You can control the columns where a change causes a history to be created. You can change the properties by accessing the Oracle Utilities Analytics Administration Tool. Once the configurations have been changed, the following script needs to be executed to ensure that the changes have been applied to the appropriate scenarios:

For Unix:

Perform the following steps:

1. Navigate to *<Install_Dir>*/bin directory.

2. Add Java 8 to the path variable as shown below:

```
export PATH=<Path_To_JDK_Directory>/bin:$PATH
```

3. Run ksh ./updateODIMetadata.sh

For Windows:

Perform the following steps:

1. Navigate to *<Install_Dir>*/bin directory.

2. Add Java 8 to the path variable as shown below:

```
set PATH= <Path_To_JDK_Directory>\bin;%PATH%
```

3. Run updateODIMetadata.cmd.

> **Important!** After the SCD2 dimension behavior is customized, the execute the script after every upgrade to retain the customized behavior.

ETL Replication View

Every ETL job in the data warehouse is based on one or more Oracle Data Integrator interfaces to populate the target fact/dimension table. The starting point is a view defined for the ETL job in replication layer of the specific source product. This ETL replication view fetches the base values from the replication tables as a starting point for further transformations to be done by the Oracle Data Integrator interfaces.

In certain business scenarios, it might be necessary for you to override the definition of the provided ETL replication view. For example, in Oracle Utilities Network Management System, a particular type of data may be stored in different tables based on the network model. The base view definitions supplied for Oracle Utilities Network Management System ETL are designed based on the OPAL network model. When you use a different model, you might have to change the definition of some of these views in the replication layer for your network model.

To take a more specific example, the B1_D_ACCT_VW for the account dimension in the Oracle Utilities Network Management System star schema tries to fetch the account related attributes from the CU_SERVICE_LOCATIONS table. If your model stores the account attributes in a different table, then the view definition needs to be changed. The Maintain ETL Replication Views page in the Oracle Utilities Analytics Administration Tool allows you to do this. You need to define a custom Oracle Data Integrator interface to generate the view.

The view name and the column names in the view should be exactly as it was delivered. However, the view definition can be changed per your requirements.

A scenario should be generated for the custom Oracle Data Integrator interface and should be shown in this Oracle Utilities Analytics Administration Tool page against the appropriate view name with a definition that needs to be overridden.

When views are then generated using the 'Run View Generator' script, the ETL framework generates the view based Oracle Data Integrator scenario that you entered.

NMS Configuration

Before attaching a specific source system to Oracle Utilities Analytics, there are some configurations that need to be done for every source system for its ETL to function properly. These configurations include setting up of certain parameters to be referred to during the extraction or setting up bucket ranges for classifying the measures in a fact.

Set up the configurations needed for the Oracle Utilities Network Management System ETL processes via the Oracle Utilities Analytics Administration Tool using the two pages grouped under the **NMS Configuration**.
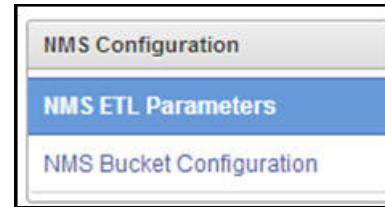
Note: Details are covered under the Configuring Source > Oracle Utilities Network Management System (NMS) sub-section under the **Chapter 5: Configuring Utilities Analytics**, where the configurations needed for every source system are discussed at length with all necessary details.

ETL Job Execution

The **Job Execution** page in Oracle Utilities Analytics Administration user interface is used for tracking the execution of the ELT processes.

An entry is created for each execution. The following screen capture shows the job execution log table.

Some attributes are populated from the SNP_SESSION table, which is used by Oracle Data Integrator to track sessions.

Attribute	Purpose
Job Configuration	A reference to the job configuration.
Session Number	Reference to the Oracle Data Integrator session number.
Scheduled Start Time	Start time as set in the schedule.
Slice Start	Starting timestamp of the slice.
Slice End	Ending timestamp of the slice.
Status	A composite status for the job. The status is primarily derived from SNP_SESSION; however, additional statuses are tracked in job executions: • Pending • Submitted • Reprocessed • Error • Done • Running
Actual Start time	The timestamp when the job actually started.
Actual End time	The timestamp when the job actually ended.
Duration	The total execution time in seconds.
Insert Count	The number of records inserted.
Update Count	The number of records updated.
Delete Count	The number of deletions.
Error Count	The number of rows identified as error.
Total Count	The number of rows processed. Sum of all the counts mentioned above.

Chapter 5

Configuring Oracle Utilities Analytics

This chapter describes how to configure, manage, and maintain an Oracle Utilities Analytics (OUA) implementation. It include:

- Configuring Source
- Configuring ELT
- Adding Sources
- Initiating the ELT Jobs
- Configuring the Analytics

Configuring Source

Before attaching a specific source system to Oracle Utilities Analytics, there are some configurations that need to be done on the source application. This section provides instructions to configure the source system prior to attaching the source instance to Oracle Utilities Analytics.

- Oracle Utilities Work and Asset Management (WAM)

- Oracle Utilities Operational Device Management (ODM)

- Oracle Utilities Customer Care & Billing (CCB)

- Oracle Utilities Network Management System (NMS)

- Oracle Utilities Mobile Workforce Management (MWM)

- Oracle Utilities Meter Data Management (MDM)

Oracle Utilities Work and Asset Management (WAM)

To enable proper data extracts for Oracle Utilities Extractors and Schema, we need to define some parameters in the Oracle Utilities Work and Asset Management application. This section provides information on the steps to be taken to enable this configuration.

- Analytics Configuration Portal

Analytics Configuration Portal

The **Analytics Configuration** portal holds information on all the analytics-oriented configuration tasks. It is a display-only portal that gives a bird's eye view of what configuration has been set up. It also provides links and guidelines for areas that require configuration to successfully run the Oracle Utilities Analytics ELT processes.

To access the **Analytics Configuration** portal in Oracle Utilities Work and Asset Management, select **Analytics Configuration** from the **Admin** menu.

Use the **WAM** tab under **BI Configuration** portal to perform the following tasks:

- BI-Extract Parameters

- Bucket Configuration

> **Note**: These configurations must be done before starting the ELT (Extract, Load, and Transform) processes on the Oracle Utilities Analytics data warehouse.

BI-Extract Parameters

During extraction of source data to the Oracle Utilities Analytics data warehouse, there are certain parameters that must be configured by implementors to identify or filter the data per business requirements. Once these parameters have been defined (by the end user), the ELT process can then use the information to selectively extract and/or transform data from the source application and populate them into the warehouse.

> **Note**: The ELT job that loads these parameters into the warehouse is configured to be initial load only. Any incremental changes to these buckets after the initial run are not captured in Oracle Utilities Analytics because any changes made to the existing parameters can cause inconsistency to the loaded data. However, if there is a need to reconfigure the parameters, the parameters can be changed on the source system and certain additional steps need to be performed to reload them into the warehouse. The star schema tables would also need to be truncated and reloaded. See the **Reloading the Extract Parameters** on page 6-8 section for the detailed steps involved in reloading the parameters.

The **BI Master Configuration BO** is delivered to contain the list of parameter values that need to be configured for use by the ELT process:

*   **Location BO Parameters:** Holds the business object names that are used to identify the locations that should be extracted for the Location Snapshot fact.

    Use this parameter configuration to specify the business objects that identify the records to be extracted into the star schema table.

*   **Asset BO Parameters:** Holds the business object names that are used to identify the assets that should be extracted for the Asset Snapshot fact.

    Use this parameter configuration to specify the business objects that identify the records to be extracted into the star schema table.

Bucket Configuration

Oracle Utilities Analytics allows you to define a set of ranges, each representing a bucket, for which extracted measures can be grouped and classified. This allows for reports that show the data as a group that matches a particular classification or bucket. For example, in Oracle Utilities Work and Asset Management, an asset's age can be classified according to an age grouping (such as Less than a Year Old, Between One to 5 Years Old, Between 5 to 10 Years Old, and Older than 10 Years).

*   Asset Age Buckets

*   Asset Installation Age Buckets

*   Asset Criticality Buckets

*   Asset Condition Score Buckets

*   Work Priority Buckets

*   Number of Cycles Overdue Buckets

*   MTBF/MTTR Buckets

    > **Note**: The ELT jobs that load these bucket configurations into the warehouse are set to be initial load only. Any incremental changes to these buckets after the initial run are not captured in Oracle Utilities Analytics because any changes made to the existing bucket configuration can cause inconsistency to the loaded fact data. However, if there is a need to reconfigure the buckets, certain additional steps need to be carried out to pull in the new configuration into the warehouse and ensure that this is reflected in the star schema tables. For the detailed steps involved in reloading the bucket configuration, see **Reloading the Bucket Configuration** on page 6-9 section.

Click the link on the **Bucket Configuration** zone to navigate to the **Bucket Configuration Maintenance** portal where the bucket values can be configured.

Asset Age Buckets

The Asset Age bucket configuration defines various bucket ranges that are used to classify how long an asset has been in the system.

No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	365	Less Than A Year
20	365	1825	1-5 Years
30	1825	3650	5 - 10 Years

Sequence	Start Range	End Range	Description
40	3650	5475	10 -15 Years
50	5475	9125	15 - 25 Years
60	9125	99999	Older Than 25 Years

This data is extracted into the Asset Age dimension in the Business Intelligence data warehouse. This is referenced in the Asset Snapshot fact to categorize an asset's age into one of these buckets.

The following rules are enforced for the buckets:

- Only one set of bucket configuration is allowed.
- No gaps or overlaps are allowed within the bucket ranges.
- One and only one catch-all bucket is allowed and required (the bucket end range has the maximum value of 99999).
- One and only one initial bucket is allowed and required (bucket range that starts at 0).

Asset Installation Age Buckets

The Asset Installation Age bucket configuration defines various bucket ranges that are used to classify how long an asset has been installed on the field.

No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	365	Less Than A Year
20	365	1825	1-5 Years
30	1825	3650	5 - 10 Years
40	3650	5475	10 -15 Years
50	5475	9125	15 - 25 Years
60	9125	99999	Older Than 25 Years

This data is extracted into the Asset Age dimension in the Business Intelligence data warehouse. This is referenced in the Asset Snapshot fact to categorize an asset's age into one of these buckets.

The following rules are enforced for the buckets:

- Only one set of bucket configuration is allowed.
- No gaps or overlaps are allowed within the bucket ranges.
- One and only one catch all bucket is allowed and required (the bucket end range has the maximum value of 99999).
- One and only one initial bucket is allowed and required (bucket range that starts at 0).

Asset Criticality Buckets

The Asset Criticality bucket configuration defines various bucket ranges that are used to classify the criticality defined at a given location. No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	31	Low Criticality
20	31	61	Medium Criticality
30	61	81	High Criticality
40	81	99999	Extremely Critical

This data is extracted into the **Location** dimension in the Business Intelligence data warehouse. The criticality value in the location dimension will be categorized into one of these buckets.

The following rules are enforced for the buckets:

- Only one set of bucket configuration is allowed.
- No gaps or overlaps are allowed within the bucket ranges.
- One and only one catch all bucket is allowed and required (the bucket end range has the maximum value of 99999).
- One and only one initial bucket is allowed and required (bucket range that starts at 0).
- The Criticality Category must be unique per bucket range.
- A bucket range with Critical category is required.

Asset Condition Score Buckets

The Asset Condition Score bucket configuration defines various bucket ranges that are used to classify the condition score of a given asset. No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	2	Very Good
20	2	3	Good
30	3	4	Average
40	4	5	Poor
50	5	99999	Extremely Critical

This data is extracted into the Asset dimension in the Business Intelligence data warehouse. The condition score value in the asset dimension will be categorized into one of these buckets.

The following rules are enforced for the buckets:

- Only one set of bucket configuration is allowed.
- No gaps or overlaps are allowed within the bucket ranges.
- One and only one catch-all bucket is allowed and required (the bucket end range has the maximum value of 99999).
- One and only one initial bucket is allowed and required (bucket range that starts at 0).

Work Priority Buckets

The Work Priority bucket configuration defines various bucket ranges that are used to classify the priority set for a work order. No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Priority Category	Description
10	0	5		Low
20	5	6		Normal
30	6	7	High Priority	High
40	7	9	High Priority	Urgent
50	9	99999	High Priority	Emergency

This data is extracted into the **Work Order** dimension in the Business Intelligence data warehouse. The priority value in the work order dimension will be categorized into one of these buckets.

The following rules are enforced for the buckets:

- Only one set of bucket configuration is allowed.

- No gaps or overlaps are allowed within the bucket ranges.

- One and only one catch all bucket is allowed and required (the bucket end range has the maximum value of 99999).

- One and only one initial bucket is allowed and required (bucket range that starts at 0).

- At least one bucket range must be configured as High Priority.

Number of Cycles Overdue Buckets

The Number of Cycles Overdue bucket configuration defines various bucket ranges that are used to classify how long an work order activity has been overdue in the system. No values are delivered as a part of the base product. Here are some sample values:

Number of Cycles	Catch All?	Description
1		1 Cycle
2		2 Cycles
3		3 Cycles
4	Yes	4+ Cycles

This data is extracted into the **Number of Cycles Overdue** dimension in the Business Intelligence data warehouse. This is referenced in the Work Order Activity fact to categorize the number of days it has been overdue into one of these buckets.

The following rules are enforced for the buckets:

- Only one set of bucket configuration is allowed.

- No gaps or overlaps are allowed within the bucket ranges.

- One and only one catch all bucket is allowed and required (the bucket end range has the maximum value of 99999).

- One and only one initial bucket is allowed and required (bucket range that starts at 0).

- One and only one row with a checked Catch All Indicator is required and allowed.

MTBF/MTTR Buckets

The MTBF/MTTR bucket configuration defines various bucket ranges that are used to determine the Mean Time Between Failure (MTBF) or Mean Time To Repair (MTTR) ratings of a given asset. No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	366	Year To Date
20	0	731	Last 2 Years
30	0	1096	Last 3 Years
40	0	1461	Last 4 Years
50	0	1826	Last 5 Years

This data is replicated into the data warehouse and is referenced in the Asset and Location Snapshot facts to determine the MTBF rating, MTTR rating, and the count of failures for each of these buckets. The calculated values will be populated into dedicated measure columns on the facts.

The following rule is enforced for the buckets:

- Only one set of the bucket configurations is allowed.

Oracle Utilities Operational Device Management (ODM)

To enable proper data extracts for Oracle Utilities Extractors and Schema, we need to define some parameters in the Oracle Utilities Operational Device Management application. This section provides information on the steps to enable this configuration.

- Analytics Configuration Portal

Analytics Configuration Portal

The **Analytics Configuration** portal holds information on all the analytics-oriented configuration tasks. It is a display-only portal that gives a bird's eye view of how much configuration has been set up, and also provides links and guidelines for the areas that need configuration, at the minimum, to successfully run the Oracle Utilities Analytics ELT processes from Oracle Utilities Analytics.

To access the **Analytics Configuration** portal in Oracle Utilities Operational Device Management, select **Analytics Configuration** from the **Admin** menu.

Use the **Analytics Configuration** portal to perform the following tasks:

- BI-Oriented Master Configuration

- Bucket Configuration

- BI-Oriented Extendable Lookups

> **Note**: These configurations must be done before starting the ELT (Extract, Load, and Transform) processes on the Oracle Utilities Analytics data warehouse.

BI-Oriented Master Configuration

This section lists the master configuration BO that was created for Oracle Utilities Extractors and Schema and guides you during the configuration. Click the link in the **Master Configuration** zone to navigate to the **Master Configuration Maintenance** portal where the configuration is maintained.

Extract Parameters

During extraction of source data to BI data warehouse, there are certain parameters that need to be configured by implementation to identify or filter the data per business requirements. Once these parameters have been set up (by the end user), the ELT process can then use this information to selectively extract data and/or transform data from the source application and populate them into the warehouse.

> **Note**: The ELT job that loads these parameters into the warehouse is configured to be initial load only. Any incremental changes to these buckets after the initial run is not captured in Oracle Utilities Analytics. Because, any changes made to the existing parameters can cause inconsistency to the loaded data. However, if a need arises to reconfigure the parameters, the parameters can be changed on the source system, and certain additional steps need to be performed to reload them into the warehouse. The star schema tables would also need to be truncated and reloaded. For the detailed steps involved in reloading the parameters, see **Reloading the Extract Parameters** on page 6-8 section.

The **BI Master Configuration BO** is delivered to contain the list of parameter values that need to be configured for use by the ELT process:

- **Firmware Parameters**: Holds the firmware identifier types that are used in retrieving and extracting the appropriate firmware versions of the asset. The Operational Device dimension in the Business Intelligence data warehouse has allocated 10 slots for storing an asset's firmware versions.

  Use this firmware parameter configuration to specify the type of firmware versions that should be extracted into each of the columns.

- **Operation Device BO Parameters**: Holds the business object names that are used to identify the operational devices that should be extracted. The Operational Device Snapshot and Operation Device Accumulation facts, and the Operational Device dimension in the Business Intelligence data warehouse are designed to hold only the operational device data.

  Use this operational device parameter configuration to specify the business objects that are used to identify the assets that should be extracted into the above mentioned star schema tables.

- **Asset Failure Parameters**: Holds the failure condition codes to be considered for identifying asset failures during the extraction process. The Operational Device Snapshot and Asset Location facts in the Business Intelligence data warehouse have specific columns that indicate an asset's failure.

  Use this asset failure parameter configuration to specify the conditions that should be considered for marking an asset's failure.

Bucket Configuration

Oracle Utilities Analytics allows you to define a set of ranges, each representing a bucket, for which extracted measures can be grouped and classified under the relevant bucket. This allows for reports that show the data as a group that matches a particular classification, or bucket. For example, in Oracle Utilities Operational Device Management, an asset's age can be classified according to an age grouping (such as Less than 6 Months Old, Between 6 Months to a Year Old, Between One to 5 Years Old, and Older than 5 Years).

- Defining Age Buckets
- Asset Age Buckets
- Asset Installation Age Buckets
- Asset In Storage Age Buckets
- Bucket Definition Considerations

Defining Age Buckets

Use the **Bucket Configuration** portal in Oracle Utilities Operational Device Management to define any one of these types of age buckets:

- Asset Age
- Asset Installation Age
- Asset In Storage Age

These buckets can be defined for a specific asset type or as a default set of buckets that is applicable to all assets that do not have any specific buckets defined on the asset type.

Click the link on the **Bucket Configuration** zone to navigate to the **Bucket Configuration Maintenance** portal where the bucket values can be configured.

> **Note**: All the ELT jobs that load these bucket configurations into the warehouse are set to be initial load only. Any incremental changes to these buckets after the initial run is not captured in Oracle Utilities Analytics. Because, any changes made to the existing bucket configuration can cause inconsistency to the loaded fact data. However, if there is a need to reconfigure the buckets, certain additional steps need to be carried out to pull in the new configuration into the warehouse and ensure that this is reflected in the star schema tables.
>
> For the detailed steps involved in reloading the bucket configuration, see Reloading the Bucket Configuration section.

Asset Age Buckets

The Asset Age bucket configuration defines various bucket ranges that are used to classify how long an asset has been in the system. See the Bucket Definition Considerations section for various rules to be followed while defining the buckets.

No values are delivered as a part of the base product. Here are some sample values:

Default: Yes

Sequence	Start Range	End Range	Description
10	0	90	Less Than 3 Months
20	90	180	3 - 6 Months
30	180	270	6 - 9 Months
40	270	365	9 - 12 Months

Sequence	Start Range	End Range	Description
50	365	730	1 - 2 Years
60	730	1095	2 - 3 Years
70	1095	99999	Older Than 3 Years

This data is extracted into the Asset Age dimension in the Business Intelligence data warehouse. This is referenced in the Operational Device Snapshot fact to categorize an asset's age into one of these buckets.

Asset Installation Age Buckets

The Asset Installation Age bucket configuration defines various bucket ranges that are used to classify how long an asset has been installed on the field. See the Bucket Definition Considerations section for various rules to be followed while defining the buckets.

No values are delivered as a part of the base product. Here are some sample values:

Default: Yes

Sequence	Start Range	End Range	Description
10	0	90	Less Than 3 Months
20	90	180	3 - 6 Months
30	180	270	6 - 9 Months
40	270	365	9 - 12 Months
50	365	730	1 - 2 Years
60	730	1095	2 - 3 Years
70	1095	99999	Older Than 3 Years

This data is extracted into the **Asset Install Age** dimension in the Business Intelligence data warehouse. This is referenced in the Operational Device Snapshot fact to categorize an asset's install age into one of these buckets.

Asset In Storage Age Buckets

The Asset In Storage Age Buckets configuration defines various bucket ranges that are used to classify how long an asset has been available in the storerooms. See the Bucket Definition Considerations section for various rules to be followed while defining the buckets.

No values are delivered as a part of the base product. Here are some sample values:

Default: Yes

Sequence	Start Range	End Range	Description
10	0	30	Less Than 1 Month
20	30	180	2 - 6 Months
30	180	365	7 - 12 Months
40	365	99999	Older Than a Year

This data is extracted into the Asset Instore Age dimension in the Business Intelligence data warehouse. This is referenced in the Operational Device Snapshot fact to categorize an asset's instore age into one of these buckets.

Bucket Definition Considerations

To define the age bucket ranges follow these rules:

- At least one bucket has the start range as 0.

- At least one bucket has the end range as 99999.

- No overlapping is allowed between buckets. For example: 0-10, 5-20 is an incorrect way to define buckets.

- No gap in allowed between buckets. For example: 0-10, 15-20 or 0-10, 11-15 is an incorrect way to define buckets.

BI-Oriented Extendable Lookups

This section lists all the extendable lookup BOs that were created for Business Intelligence, and guides you to set up the lookup values. Click the link on the **Extendable Lookup** zone to navigate to the **Extendable Lookup Maintenance** portal where the lookup values are configured.

DSI Identifier Mapping

The data source indicator (DSI) identifier mapping defines the integration points of Oracle Utilities Operational Device Management with other Oracle Utilities products (such as Oracle Utilities Meter Data Management). This information is pulled into the BI data warehouse to be used by the ELT processes. During the actual ELT, Oracle Utilities Operational Device Management facts are joined with Oracle Utilities Meter Data Management dimensions using the DSI identifier mapping information.

The Business Intelligence data warehouse uses Service Point integration between Oracle Utilities Operational Device Management with Oracle Utilities Meter Data Management to join Oracle Utilities Operational Device Management facts and Oracle Utilities Meter Data Management / Oracle Utilities Customer Care and Billing dimensions. Use the DSI identifier mapping to configure the following:

- **Maintenance Object**: The maintenance object of the entity being integrated between Oracle Utilities Operational Device Management and the external system.

  Note: In the OUASA 2.4.1 release, the ELT processes extract DSI mapping for 'Node' maintenance object from the source Oracle Utilities Operational Device Management system.

- **External System Mapping**: Configure the following items under this group:

  - **Identifier Type**: The identifier type code used to identify the external identifier number from the integrating system.

  - **Product**: The product to which Oracle Utilities Operational Device Management is being integrated with.

  - **Data Source Indicator**: The unique identifier of the integrating system. This is the environment ID of the Oracle Utilities Meter Data Management system. It can be identified from the Oracle Utilities Meter Data Management system using the below query.

    ```
    select ENV_ID from F1_INSTALLATION;
    ```

Oracle Utilities Customer Care & Billing (CCB)

To enable proper data extracts for Oracle Utilities Extractors and Schema, certain parameters have to be defined in the Oracle Utilities Customer Care and Billing application. This section provides information on the steps to be performed to enable this configuration.

• Analytics Configuration Portal

Analytics Configuration Portal

To enable successful data extracts for Oracle Utilities Extractors and Schema, certain configurations are required in the Oracle Utilities Customer Care and Billing application. This section provides information on what are the needed configuration in Oracle Utilities Customer Care and Billing and how to properly set them up.

To access the **Analytics Configuration** portal in Oracle Utilities Customer Care and Billing, select **Analytics Configuration** from the **Admin Menu**.

Use the **Analytics Configuration** portal to perform the following tasks:

• Analytics-Oriented Master Configuration

• Bucket Configuration

> **Note**: These configurations must be done before starting the ELT processes on the Oracle Utilities Analytics data warehouse.

Analytics-Oriented Master Configuration

This section lists the master configuration BO that was created for Oracle Utilities Extractors and Schema, and guides you during the configuration. The **Analytics Configuration** Portal provides a link that allows you to navigate to the **Master Configuration** portal, where master configurations are maintained.

Extract Parameters

During extraction of source data to BI data warehouse, there are certain parameters that need to be configured by implementation to identify or filter the data per business requirements. Once these parameters have been set up (by the end user), the ELT process can then use these information to selectively extract data and/or transform data from the source application and populate them into the warehouse.

> **Note**: The ELT job that loads these parameters into the warehouse is configured to be initial load only. Any incremental changes to these buckets after the initial run is not captured in Oracle Utilities Analytics. Because, any changes made to the existing parameters can cause inconsistency to the loaded data. However, if a need arises to reconfigure the parameters, the parameters can be changed on the source system, and certain additional steps need to be performed to reload them into the warehouse. The star schema tables would also need to be truncated and reloaded. For the detailed steps involved in reloading the parameters, see **Reloading the Extract Parameters** on page 6-8 section.

The BI Extract Parameters Configuration BO is delivered to contain the list of parameter values that need to be configured for use by the ELT process:

• **To Do Characteristic Types**: Used for configuring the Account, Service Agreement, Person, and Premise characteristic types that are used to determine the corresponding entities related to a To Do. This is used for the To Do facts.

• **Recent To Do Miscellaneous**: Used for configuring a numeric value that denotes the 'Completed in Days' parameter. Completed in Days is used to retrieve the To Do entries completed within the last 'x' days, for inclusion on the Recent To Do Fact.

- **Financial Transaction Characteristic Values**: Holds the Revenue and Tax characteristic values that are used to determine the Revenue and Tax amount, respectively, of the FT. These values are used by the Financial Transaction, Financial Transaction General Ledger, and SA Billing extracts.

- **Financial Transaction Characteristic Types**: Holds the GL Account characteristic types that are used to determine the GL Accounts that are extracted for the Financial Transactions, Financial Transaction General Ledger, and SA Billing extracts.

- **Exclude Archive Adjustment Types**: Holds all the archive adjustment types that should be excluded by the FT and FT GL extracts.

- **Broken PA Characteristic Types**: Holds the Broken characteristic type that is used to identify if a payment arrangement is broken. This is used by the Payment Arrangement extract.

- **Broken PA Characteristic Values**: Holds the Broken characteristic value that is used to identify if a payment arrangement is broken. This is used by the Payment Arrangement extract.

- **Broken PA Adjustment Cancel Reasons**: Holds the adjustment cancel reason that is used to identify if a payment arrangement is broken. This is used by the Payment Arrangement extract.

- **Overdue Bill Characteristic Types**: Holds the overdue bill characteristic type that is used to identify the bill that is associated with the Overdue Process. This is used by the Overdue Process and Event extracts.

- **High Bill Complaint Characteristic Types**: Holds the characteristic type that is used to identify the bill associated with the High Bill Complaint case. This is used by the SA Billing extract.

- **Exclude High Bill Complaint Case States**: Holds the case statuses for which High Bill Complaint cases should not be considered. This is used by the SA Billing extract to identify if a bill is associated to a high bill complaint.

- **High Bill Complaint Case Types**: Holds the case types for high bill complaint cases. This is used by the SA Billing extract.

- **SA Snapshot Miscellaneous**: Holds a numeric value to denote the Exclude SAs Closed in Days parameter. Exclude SAs Closed in Days, if specified, is used to retrieve SAs closed within the last 'x' days, for inclusion on the SA Snapshot and Payment Arrangement Snapshot extracts.

Bucket Configuration

Oracle Utilities Analytics allows you to define a set of ranges, each representing a bucket, for which extracted measures can be grouped and classified under the relevant bucket. This allows for reports that show data as a group that matches a particular classification, or bucket. For example, in Oracle Utilities Customer Care and Billing, arrears can be classified according to their arrears age (For example, Less Than 30 Days, 30-60 Days, 6-90 Days, Older Than 90 Days).

Defining Age Buckets

The **Analytics Configuration** portal in Oracle Utilities Customer Care and Billing has a **Bucket Configuration List** zone that lists all the buckets that need to be configured for business intelligence:

- Billing Day In Window Buckets

- Days Before Bill Window Closes Buckets

- Days of Unbilled Usage Buckets

- Days Since Last Frozen Bill Segment Buckets

- PA Future Payment Age Buckets

- PA Number of Installments Buckets

- PA Recurring Charge Amount BucketsPA Recurring Charge Amount Buckets

- PP Future Payment Age Buckets

- SA Arrears Buckets

The **Bucket Configuration List** zone provides an overview of which bucket configurations have been set up so far, and which are yet to be set up. Those that have not been set up yet will have a corresponding **Add** button which, when clicked, will navigate you to the **Bucket Configuration Maintenance** portal in **Add** mode. Those that have been previously set up will have a foreign key reference link which, when clicked, will navigate you to the **Bucket Configuration Maintenance** portal where you can view or edit the existing bucket values.

> **Note**: All the ELT jobs that load these bucket configurations into the warehouse are set to be initial load only. Any incremental changes to these buckets after the initial run is not captured in Oracle Utilities Analytics. Because, any changes made to the existing bucket configuration can cause inconsistency to the loaded fact data. However, if there is a need to reconfigure the buckets, certain additional steps need to be carried out to pull in the new configuration into the warehouse and ensure that this is reflected in the star schema tables.
>
> For the detailed steps involved in reloading the bucket configuration, see Reloading the Bucket Configuration section.

Billing Day In Window Buckets

The Billing Day In Window bucket configuration defines the bucket ranges that identify the day in the billing window when the bill segment was frozen. It also indicates whether the range is within the window or outside the window. Two sets of ranges can be defined for this bucket configuration: one set for day ranges that fall inside the billing window (bill window status of In Window), and another set for day ranges that fall outside the billing window (bill window status of Outside Window).

No values are delivered as a part of the base product. Here are some sample values:

Sample 1: Bill Window Status: In Window

Sequence	Bill Window Category	Start Range	End Range	Description
10	First Day	0	1	Day 1
20		1	2	Day 2
30		2	3	Day 3
40		3	4	Day 4
50		4	5	Day 5
60		5	10	Day 5+

Sample 2: Bill Window Status: Outside Window

Sequence	Bill Window Category	Start Range	End Range	Description
10		0	99,999	Outside Billing Window

This data is extracted into the Billing Day In Window dimension in the Business Intelligence data warehouse. This is referenced in the SA Billing fact to determine when in the billing window was the bill segment frozen.

The billing day in window buckets enforce the following rules:

- Only one set of ranges can be defined for each bill window status (In Window and Outside Window).

- No gaps or overlaps are allowed within the bucket ranges for a bill window status.

- One initial bucket is required (bucket range starts at 0).

- One catch all bucket is required for the bucket configuration with the bill window status of Outside Window (bucket end range has the maxim value of 99999).

- One and only one bucket range should be categorized as "First Day" for the bucket configuration with the bill window status of In Window.

Days Before Bill Window Closes Buckets

The Days Before Bill Window Closes bucket configuration defines the bucket ranges that identify the number of days left before the bill window closes. Two sets of ranges can be configured for this bucket configuration: one set for day ranges when the bill window is still open (window closure status is Open), and another set for days when the bill window is closed (window closure status is Closed).

No values are delivered as a part of the base product. Here are some sample values:

Sample 1: Window Closure Status: Open

Sequence	Start Range	End Range	Description
10	0	1	1 Day Before Bill Window Closes
20	1	2	2 Days Before Bill Window Closes
30	2	3	3 Days Before Bill Window Closes
40	3	4	4 Days Before Bill Window Closes
50	4	99,999	5+ Days Before Bill Window Closes

Sample 2: Window Closure Status: Closed

Sequence	Start Range	End Range	Description
10	0	99,999	Bill Window is Closed

This data is extracted into the Days Before Bill Window Closes dimension in the Business Intelligence data warehouse. This is referenced in the SA Billing fact to determine, for bill segments in Error state, the number of days that are left for correcting the bill segment before the bill window closes.

- The days before bill window closes bucket configuration enforces the following rules:

- Only one set of bucket configuration can be defined for each window closure status (Open and Closed)

- No gaps or overlaps are allowed within the bucket ranges of a window closure status.

- Only one range is allowed for the closed window status and it has to catch all the measures (end range has the maximum value of 99999).

Days of Unbilled Usage Buckets

The Days of Unbilled Usage bucket configuration defines the bucket ranges used to identify how many days the service agreement has usage but has not been billed for.

No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	30	0 - 30 Days
20	30	60	30 - 60 Days
30	60	90	60 - 90 Days
40	90	99,999	90+ Days

This data is extracted into the Days of Unbilled Usage dimension in the Business Intelligence data warehouse. This is referenced in the SA Arrears Snapshot fact to determine how many days that the service agreement has usage but has not been billed.

The following rules are enforced for days of unbilled usage buckets:

- Only one set of the days of unbilled usage bucket configuration is allowed.

- No gaps or overlaps are allowed within the bucket ranges.

- One and only one catch all bucket is allowed and required (the bucket end range has the maximum value of 99999).

- One and only one initial bucket is allowed and required (bucket range that starts at 0).

Days Since Last Frozen Bill Segment Buckets

The Days Since Last Frozen Bill Segment bucket configuration defines the bucket ranges used to identify the number of days since the service agreement has a frozen bill segment.

No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	30	0 - 30 Days
20	30	60	30 - 60 Days
30	60	90	60 - 90 Days
40	90	180	90 - 180 Days
50	180	99,999	180+ Days

This data is extracted into the Days Since Last Frozen Bill Segment dimension in the Business Intelligence data warehouse. This is referenced in the SA Arrears Snapshot fact to determine how long it has been since the service agreement has frozen bill segment.

The following rules are enforced for days since last frozen bill segment buckets:

- Only one set of the days since last frozen bill segment bucket configuration is allowed.

- No gaps or overlaps are allowed within the bucket ranges.

- One and only one catch all bucket is allowed and required (the bucket end range has the maximum value of 99999).

- One and only one initial bucket is allowed and required (bucket range starts at 0).

PA Future Payment Age Buckets

The PA Future Payment Age defines the configurable day ranges indicating the age of future scheduled payments for payment arrangements. The future payments for a payment arrangement are approximated using the number of installments left and the frequency that the payment arrangement SA is billed (via the bill cycle schedule of the account).

No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	30	30 Days
20	30	60	30 - 60 Days
30	60	90	60 - 90 Days
40	90	99,999	90+ Days

This data is replicated into the data warehouse and is referenced in the PA Accumulation/ Snapshot fact to categorize a payment arrangement's future payments into one of these age buckets.

The following rules are enforced for PA future payment age buckets:

- Only one set of future payment age bucket configuration is allowed.

- No gaps or overlaps are allowed within the bucket ranges.

- One and only one catch all bucket is allowed and required (bucket end range has maximum value of 99999).

PA Number of Installments Buckets

The PA Number of Installments bucket defines the bucket ranges used to group payment arrangements based on the number of installments.

No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	3	< 3 Installations
20	3	5	3 - 5 Installations
30	5	8	6 - 8 Installations
40	8	99,999	8+ Installations

This data is extracted into the PA Number of Installments dimension in the Business Intelligence data warehouse. This is referenced in the PA Accumulation/Snapshot fact to determine which bucket a payment arrangement falls into based on its number of installments.

The following rules are enforced on the PA number of installments buckets:

- Only one set of number of installment bucket configuration is allowed.

- No gaps or overlaps are allowed within the bucket ranges.

- One and only one catch all bucket is allowed and required (bucket end range has maximum value of 99999).

- One and only one initial bucket is allowed and required (bucket range starts at 0).

PA Recurring Charge Amount Buckets

The PA Recurring Charge Amount bucket defines the bucket ranges used to group payment arrangements based on their recurring charge amounts.

.No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	25	Less than $25
20	25	50	$25 - $50
30	50	100	$50 - $100
40	100	200	$100 - $200
50	200	99,999	Greater than $200

This data is extracted into the PA Recurring Charge Amount dimension in the Business Intelligence data warehouse. This is referenced in the PA Accumulation/Snapshot fact to determine which bucket a payment arrangement falls into based on its recurring charge amount.

The following rules are enforced for recurring charge amount buckets:

• Only one set of recurring charge amount bucket configuration is allowed.

• No gaps or overlaps are allowed within the bucket ranges.

• One and only one catch all bucket is allowed and required (bucket end range has maximum value of 99999).

• One and only initial bucket is allowed and required (bucket range starts at 0).

PP Future Payment Age Buckets

The PP Future Payment Age bucket defines the age buckets for which future payments for pay plans are scheduled to be paid.

No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	30	0 - 30 Days
20	30	60	30 - 60 Days
30	60	90	60 - 90 Days
40	90	99,999	90+ Days

This data is replicated into the data warehouse and referenced in the Pay Plan Accumulation/Snapshot fact to categorize a payment plan's future payments into one of these buckets.

The following rules are enforced on the pay plan future payment age buckets:

• Only one set of future payment age bucket configuration is allowed.

• No gaps or overlaps are allowed within the bucket ranges.

• One and only one catch all bucket is allowed and required (bucket end range has maximum value of 99999).

SA Arrears Buckets

The SA Arrears bucket defines the bucket ranges for the service agreement's arrears age.

Sequence	Start Range	End Range	Description
10	0	15	0 - 15 Days
20	15	31	15 - 30 Days
30	31	61	31 - 60 Days
40	61	91	61 - 90 Days
50	91	99,999	90+ Days
60	31	99,999	31+ Days

This data is replicated into the data warehouse and is referenced in the SA Arrears Snapshot fact to categorize a service agreement's arrears into one of these buckets based on its age.

The following rules are enforced for SA arrears buckets:

•   Only one set of arrears age bucket configuration is allowed.

•   At least one catch all bucket has to be defined (bucket end range has maximum value of 99999).

Oracle Utilities Network Management System (NMS)

To enable successful data extracts for Oracle Utilities Extractors and Schema, certain configurations are required in the Oracle Utilities Analytics Administration Tool and Oracle Utilities Network Management System (NMS).

The following sections provide information on how to set up this configuration:

•   Configuring Oracle Utilities Analytics

•   Configuring Oracle Utilities Network Management System

Configuring Oracle Utilities Analytics

This section provides information about the various configurations to be setup so that the Oracle Utilities Network Management System's ELT functions successfully. Unlike other source systems, these configurations have to be setup using the Oracle Utilities Analytics Administration Tool.

ELT Parameters

During extraction of source data to BI data warehouse, there are certain parameters that need to be configured by implementation to identify or filter the data per business requirements. For example, in order to identify a particular outage event as a momentary outage, implementation has to define the duration of an outage so that it is considered momentary. Once these parameters have been set up (by the end user), the ELT process can then use these information to selectively extract data and/or transform data from the source application and populate them into the warehouse.

> **Note**: The ELT job that loads these parameters into the warehouse is configured to be initial load only. Any incremental changes to these buckets after the initial run is not captured in Oracle Utilities Analytics. Because, any changes made to the existing parameters can cause inconsistency to the loaded data. However, if a need arises to reconfigure the parameters, the parameters can be changed on the source system, and certain additional steps need to be performed to reload them into the warehouse. The star schema tables would also need to be truncated and reloaded. For the detailed steps involved in reloading the parameters, see Reloading the Extract Parameters.

The various types of parameters that need to be configured for Oracle Utilities Network Management System ELT are as follows:

- **Country**: Country code where the addresses are located.

- **Interruptions Count for CEMI**: Number of interruptions that you should have experienced to calculate the CEMI index. This is required by the Control Zone Outage and City Outage extracts.

- **Momentary Outage Duration**: Duration in minutes to be used for determining if an outage is a momentary outage or a sustained outage. This is required by the Control Zone Outage and City Outage extracts.

- **Number of Retention Days**: Number of days in which restored outage data is retained in the Recent Call, Recent Crew, Recent Job and Recent Customer Outage facts.

- **Priority Indicator Bit**: Nth bit, which indicates the priority of the Complaint registered in the Incidents table required by the Call fact extracts.

- **Switch Plan Aborted Step Status Key(s)**: Codes used to identify the aborted steps status of a Switch Plan. This is required by the Switch Plan extracts.

- **Switch Plan Begin State Key(s)**: Codes used to identify the initial states of Switch Plans.

- **Switch Plan End State Key(s)**: Codes used to identify the final states of Switch Plans.

- **Switch Plan Failed Step Status Key(s)**: Codes used to identify the failed steps states of a Switch Plan. This is required by the Switch Plan extracts.

- **Switch Sheet Planned Category**: Planned Category code for Switch Sheets.

- **Event Status:** Holds a list of excluded Event Status values. Those event status values specified in the parameters are excluded from the extract. Multiple values can be configured for this parameter.

- **Account Type:** Holds a list of excluded Account Type values. Those Account Type values specified in the parameters are excluded from the extract. Multiple values can be configured for this parameter.

Bucket Configuration

Oracle Utilities Analytics allows you to define a set of ranges, each representing a bucket, for which extracted measurement values can be grouped and placed against the relevant bucket. This allows analysts to view the data as a group that matches a particular classification, or bucket. For example, in Oracle Utilities Network Management System, analysts may want to classify the outages based on certain ranges of duration.

> **Note**: All the ELT jobs that load these bucket configurations into the warehouse are set to be initial load only. Any incremental changes to these buckets after the initial run is not captured in Oracle Utilities Analytics. Because, any changes made to the existing bucket configuration can cause inconsistency to the loaded fact data. However, if there is a need to reconfigure the buckets, certain additional steps need to be carried out to pull in the new configuration into the warehouse and ensure that this is reflected in the star schema tables. For the detailed steps involved in reloading the bucket configuration, see the Reloading the Bucket Configuration section.

There are three types of buckets that need to be configured for the source system Oracle Utilities Network Management System.

- Outage Duration Buckets

- Estimated Restoration Duration Buckets

- Duration Deviation Buckets

- Customers Out Buckets

Outage Duration Buckets

The outage duration bucket configuration defines the bucket ranges used for classifying each outage based on how long it took before the outage was addressed and closed. The restored customer outage fact uses this bucket configuration to map the outage duration measure to the corresponding bucket range. The configured data is extracted into the Outage Duration dimension via a separate ELT job.

No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	30	Restored within 30 minutes
20	30	60	30 minutes to 1 hour
30	60	9999999	More than an hour for restoration

The following rules are enforced for outage duration buckets:

- No gaps or overlaps are allowed within the bucket ranges.

- One catch all bucket should be defined (the bucket end range has maximum value of 9999999).

Estimated Restoration Duration Buckets

The estimated restoration duration bucket configuration defines the bucket ranges used for classifying outages based on its estimated restoration time. The restored customer outage fact uses this bucket configuration to map the estimated restoration duration measure to the corresponding bucket range. The configured data is extracted into the Estimated Restoration Duration dimension via a separate ELT job.

No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	30	To be restored within 30 minutes
20	30	60	30 minutes to 1 hour
30	60	9999999	More than an hour for restoration

The following rules are enforced for estimated restoration duration buckets:

- No gaps or overlaps are allowed within the bucket ranges.

- One catch all bucket should be defined (the bucket end range has maximum value of 9999999).

Duration Deviation Buckets

The duration deviation bucket configuration defines the bucket ranges used for classifying an outage based on how much its actual restoration time deviated from its estimated restoration time. The restored customer outage fact uses this bucket configuration to map the duration deviation measure to the corresponding bucket range. The configured data is extracted into the Duration Deviation dimension via a separate ELT job.

No values are delivered as a part of the base product. Here are some sample values:

Sequence	Start Range	End Range	Description
10	0	30	0 to 30 minutes delay
20	30	60	30 minutes to 1 hour delay
30	60	120	1 to 1.5 hours delay
40	120	9999999	More than 2 hours delay

The following rules are enforced for the duration deviation buckets:

- No gaps or overlaps are allowed within the bucket ranges.

- One catch all bucket should be defined (the bucket end range has maximum value of 99999999).

Customers Out Buckets

The Customers Out bucket configuration defines the bucket range for classifying the number of customers that were out during an outage event. The configured data is extracted into the Customers Out bucket dimension via a separate ELT job.

No values are delivered as a part of the base product. Here are some sample values::

Sequence	Start Range	End Range	Description
10	0	10	0 to 10 customers out
20	10	50	10 to 50 customers out
30	50	100	50 to 100 customers out
40	100	9999999	100 to 9999999 customers out

The following rules are enforced for the duration deviation buckets:

- No gaps or overlaps are allowed within the bucket ranges.

- One catch all bucket should be defined (the bucket end range has maximum value of 99999999).

Configuring Oracle Utilities Network Management System

An entry for 'Data Source Indicator', a unique identifier for each Oracle Utilities Network Management System instance has to be configured in Oracle Utilities Network Management System. The data source indicator is used to uniquely identify the exact source of the data in the warehouse. Each instance should have a different configuration value for Data Source Indicator. Below is a sample Insert statement:

```
INSERT INTO CES_PARAMETERS (APP, ATTRIB, VALUE) VALUES ('BI',
'DATA_SOURCE_INDICATOR','000004');
```

Oracle Utilities Mobile Workforce Management (MWM)

To enable successful data extracts for Oracle Utilities Extractors and Schema, certain configurations are required in the Oracle Utilities Mobile Workforce Management (MWM) application. This chapter provides information on what are the needed configurations in Oracle Utilities Mobile Workforce Management and how to properly set them up.

• Analytics Configuration Portal

Analytics Configuration Portal

Oracle Utilities Mobile Workforce Management provides a **Analytics Configuration** portal that holds all the analytics-oriented configuration tasks. There are two configurations that are currently supported in Oracle Utilities Mobile Workforce Management: one for the new **Oracle Data Integrator-based ELT** (via the **Main** tab) and another one for the Oracle Warehouse Builder-based ETL (via the **Oracle Warehouse Builder-Based ETL** tab). Starting with Oracle Utilities Analytics v2.6, the ELT methodology uses Oracle Data Integrator therefore configuration should be made on the Main tab.

The **Analytics Configuration** portal is a display-only portal that gives a bird's eye view of the configuration set up for Oracle Utilities Extractors and Schema, and also provides links and guidelines for the areas that need configuration, at the minimum, to successfully run the ELT processes from Oracle Utilities Analytics.

To access the **Analytics Configuration** portal in Oracle Utilities Mobile Workforce Management, select **Analytics Configuration** from the **Admin** menu.

Use the **Analytics Configuration** portal to perform the following tasks:

• Bucket Configuration

• BI-Oriented Business Object List

> **Note**: These configurations must be done before starting the ELT processes on the Oracle Utilities Analytics data warehouse.

Bucket Configuration

Oracle Utilities Analytics allows you to define a set of ranges, each representing a bucket, for which extracted measurement values can be grouped and classified under the relevant bucket. This allows for reports that show the data as a group that matches a particular classification, or bucket. For example, in Oracle Utilities Mobile Workforce Management, appointments can be classified if the crew missed the appointment, and if so, by how much time.

The **Analytics Configuration** portal in Oracle Utilities Mobile Workforce Management has a Bucket Configuration List zone that lists all the buckets that need to be configured for analytics:

• Appointment Time Buckets

• Appointment Time of Day Buckets

• Late Logon Buckets

• Early Logoff Buckets

• Work Duration Deviation Buckets

• Travel Duration Deviation Buckets

• Travel Distance Deviation Buckets

• Response Time Deviation Buckets

The **Bucket Configuration List** zone provides an overview of which bucket configurations have been set up so far, and which are yet to be set up. Those that have not been set up yet will have a corresponding **Add** button which, when clicked, will navigate you to the **Bucket Configuration Maintenance** portal in **Add** mode. Those that have been previously set up will have a foreign key

reference link which, when clicked, will navigate you to the **Bucket Configuration Maintenance** portal where you can view or edit the existing bucket values.

> **Note**: All the ELT jobs that load these bucket configurations into the warehouse are set to be initial load only. Any incremental changes to these buckets after the initial run is not captured in Oracle Utilities Analytics. Because, any changes made to the existing bucket configuration can cause inconsistency to the loaded fact data. However, if there is a need to reconfigure the buckets, certain additional steps need to be carried out to pull in the new configuration into the warehouse and ensure that this is reflected in the star schema tables. For the detailed steps involved in reloading the bucket configuration, see **Reloading the Bucket Configuration** on page 6-9 section.

Appointment Time Buckets

The appointment time bucket configuration defines the time ranges (in minutes) indicating whether the crew missed an appointment and if so, by how much time. Each bucket range has an appointment state, which is used to indicate whether the appointment was met on time or if it was missed.

No values are delivered as a part of the base product. Here are some sample values:

Start Range	End Range	Appointment Status	Description
-99999	0	On Time	On Time
0	5	On Time	1 – 5 minutes late
5	15	Late	5 – 15 minutes late
15	30	Late	15 – 30 minutes late
30	99999	Late	More than 30 minutes late

The configured data is extracted into the Appointment Time dimension via a separate ELT job. The Crew Task fact and Field Activity Fact use these bucket configurations to determine if appointments are missed and by how much.

The following rules are enforced for appointment time buckets:

• Only one set of the appointment time bucket configuration is allowed.

• No gaps or overlaps are allowed within the bucket ranges.

Appointment Time of Day Buckets

The appointment time of day bucket configuration defines the time ranges (in minutes) classifying an appointment time of day.

No values are delivered as a part of the base product. Here are some sample values:

Start Range	End Range	Description
420	720	7:00am – 12:00pm
720	1080	12:00pm – 6:00pm
420	1080	All Day 7:00am – 6:00pm
0	99999	Other

The configured data is extracted into the Appointment Time of Day dimension via a separate ELT job. The Crew Task Fact and Field Activity Fact use these bucket configurations to map the appointment time to the corresponding bucket range.

The following rules are enforced for appointment time of day buckets:

• Only one set of the appointment time of day bucket configuration is allowed.

Late Logon Buckets

The late logon bucket configuration defines the time ranges (in minutes) indicating by how much time the crew is late to logon for a shift. Each bucket range has an indicator used for determining if the logon is on time or late.

No values are delivered as a part of the base product. Here are some sample values:

Start Range	End Range	On Time Indicator	Description
-99999	0	Yes	On Time
0	4	Yes	On Time
4	15	No	15 Min Late
15	30	No	30 Min Late
30	99999	No	Really Late

The configured data is extracted into the Late Logon dimension via a separate ELT job. The completed shift fact uses these bucket configurations to determine if the crew logged in late for the shift and by how much time.

The following rules are enforced for late logon buckets:

• Only one set of the late logon bucket configuration is allowed.

• No gaps or overlaps are allowed within the bucket ranges.

Early Logoff Buckets

The early logoff bucket configuration defines the time ranges (in minutes) indicating by how much time the crew is logging off earlier than planned for a shift. Each bucket range has an indicator used for determining if the logoff is on time or early.

No values are delivered as a part of the base product. Here are some sample values:

Start Range	End Range	On Time Indicator	Description
-99999	0	Yes	On Time
0	4	Yes	On Time
4	15	No	15 Min Early
15	30	No	30 Min Early
30	99999	No	Really Early

The configured data is extracted into the Early Logoff dimension via a separate ELT job. The completed shift fact use these bucket configurations to map the early logoff of a crew to the corresponding bucket range.

The following rules are enforced for early logoff buckets:

• Only one set of the early logoff bucket configuration is allowed.

• No gaps or overlaps are allowed within the bucket ranges.

Work Duration Deviation Buckets

The work duration deviation bucket configuration defines the percentage ranges to indicate the deviation of actual work duration relative to estimated work duration.

No values are delivered as a part of the base product. Here are some sample values:

Start Range	End Range	Description
-99999	-25	> 25% Under
-25	-10	10 – 25% Under
-10	-1	1-10% Under
-1	0	As Estimated
0	10	1 – 10% Over
10	25	10 – 25% Over
25	99999	> 25 % Over

The configured data is extracted into the Work Duration Deviation dimension via a separate ELT job. The Field Activity fact uses these bucket configurations determine the percentage of how much the actual work duration is equal, under, or over the estimated duration.

The following rules are enforced for work duration deviation buckets:

• Only one set of the work duration deviation bucket configuration is allowed.

• No gaps or overlaps are allowed within the bucket ranges.

Travel Duration Deviation Buckets

The travel duration deviation bucket configuration defines the percentage ranges to indicate the deviation of actual travel duration relative to estimated travel duration.

No values are delivered as a part of the base product. Here are some sample values:

Start Range	End Range	Description
-99999	-25	> 25% Under
-25	-10	10 – 25% Under
-10	-1	1-10% Under
-1	0	As Estimated
0	10	1 – 10% Over
10	25	10 – 25% Over
25	99999	> 25 % Over

The configured data is extracted into the Travel Duration Deviation dimension via a separate ELT job. The Crew Task fact uses these bucket configurations to determine the percentage of how much the actual travel duration is equal, under or over the estimated duration.

The following rules are enforced for travel duration deviation buckets:

• Only one set of travel duration deviation bucket configuration is allowed.

• No gaps or overlaps are allowed within the bucket ranges.

Travel Distance Deviation Buckets

The travel distance deviation bucket configuration defines the percentage ranges to indicate the deviation of actual travel distance relative to estimated travel distance.

No values are delivered as a part of the base product. Here are some sample values:

Start Range	End Range	Description
-99999	-25	> 25% Under
-25	-10	10 – 25% Under
-10	-1	1-10% Under
-1	0	As Estimated
0	10	1 – 10% Over
10	25	10 – 25% Over
25	99999	> 25 % Over

The configured data is extracted into the Travel Distance Deviation dimension via a separate ELT job. The Completed Shift fact uses these bucket configurations to determine how much the actual distance traveled by the crew deviated from the estimated distance.

The following rules are enforced for travel distance deviation buckets:

• Only one offset of travel distance deviation bucket configuration is allowed.

• No gaps or overlaps are allowed within the bucket ranges.

Response Time Deviation Buckets

The response time deviation bucket configuration defines ranges (in minutes) to indicate the deviation of actual response time relative to response time SLA. Each bucket has an indicator used for determining if the SLA is met or missed.

No values are delivered as a part of the base product. Here are some sample values:

Start Range	End Range	SLA Status	Description
-99999	-25	Met	> 25% Under
-25	-10	Met	10 – 25% Under
-10	-1	Met	1-10% Under
-1	0	Met	Meets SLA
0	10	Missed	1 – 10% Over

Start Range	End Range	SLA Status	Description
10	25	Missed	10 – 25% Over
25	99999	Missed	> 25 % Over

The configured data is extracted into the Response Time Deviation dimension via a separate ELT job. The Field Activity fact uses these bucket configurations to determine how much the actual response time deviated from the SLA duration.

The following rules are enforced fro response time deviation buckets:

• Only one set of response time deviation bucket configuration is allowed.

• No gaps or overlaps are allowed within the bucket ranges.

BI-Oriented Business Object List

The BI-Oriented Business Object List zone provides a list of records that need to be created to support a successful BI extract:

• Crew Time Usage

• Crew Hierarchy

• Service Area Hierarchy

Crew Time Usage

Crew Time Usage is an extendable lookup defining crew time usage types (Work, Travel, POU, Break, Depot, Idle, Logon, Logoff, Out of Service, Wait), where each crew time usage type is further classified as either productive or not. In the following examples, Travel is defined as "Not Productive" and Work is defined as "Productive".

The Crew Time Usage is used to support KPIs that show how much time crews spend on productive versus non-productive tasks, as well as KPIs that show how much time crews spend on performing logon and logoff procedures at the start and end of their shift, traveling to work, waiting to start activities, performing activities, taking breaks and handling events that cause them to be out of service.

Crew Hierarchy

Crew hierarchies organize crews into logical structures of up to three levels that represent their position relative to other crews. For example, your business practices may require crew hierarchy levels of Business Unit, Department and Office with different crews belonging to each level.

The hierarchies where the crew belongs to is included in the Crew dimension extract so that reports involving crews can be sliced and diced using this information.

> **Note**: For more information about crew hierarchies and how to set them up, refer to the **Chapter 8: Hierarchies** in *Oracle Utilities Analytics Mobile Workforce Management Configuration Guide*.

Service Area Hierarchy

Service Area hierarchies organize service areas into logical structures of up to three levels that represent their position relative to other service areas. For example, your business practices may require service area hierarchy levels of Division, District and County.

The hierarchies where the service area belongs to is included in the Service Area dimension extract so that reports involving service areas can be sliced and diced using this information.

> **Note**: For more information about service area hierarchies and how to set them up, refer to the **Chapter 8: Hierarchies** in *Oracle Utilities Analytics Mobile Workforce Management Configuration Guide*.

Oracle Utilities Meter Data Management (MDM)

To enable successful data extracts for Oracle Utilities Extractors and Schema, certain configurations are required in Oracle Utilities Meter Data Management (MDM) application. This chapter provides information on what are the needed configurations in Oracle Utilities Meter Data Management and how to properly set them up.

This section provides information about the following:

- Analytics Configuration Portal
- BI Aggregators
- Setting Up the Dynamic Usage Group Extracts
- Setting Up Snapshot Fact Extracts

Analytics Configuration Portal

Oracle Utilities Meter Data Management provides a **Analytics Configuration** portal that holds all the BI-oriented configuration tasks. There are two configurations that are currently supported in Oracle Utilities Meter Data Management: one for the Oracle Data Integrator-based ELT (via the Main tab) and another one for the Oracle Warehouse Builder-based ETL (via the Oracle Warehouse Builder-Based ETL tab). Starting with Oracle Utilities Analytics v2.6, the ELT methodology uses Oracle Data Integrator therefore configuration should be made on the Main tab.

The **Analytics Configuration** portal is a display-only portal that gives a bird's eye view of how much configuration has been set up for Oracle Utilities Extractors and Schema, and also provides links and guidelines for the areas that need configuration, at the minimum, to successfully run the data extraction process in Oracle Utilities Meter Data Management.

To access the **BI Configuration** portal in Oracle Utilities Meter Data Management, select **Analytics Configuration** from the **Admin** menu.

Use the **Analytics Configuration** portal to set up the following:

- BI-Oriented Master Configuration
- BI-Oriented Extendable Lookups
- Service Point Configuration

BI-Oriented Master Configuration

During extraction of source data to BI data warehouse, there are certain parameters that need to be configured by implementation to identify or filter the data per business requirements. Once these parameters have been set up (by the end user), the ELT process can then use these information to selectively extract data and/or transform data from the source application and populate them into the warehouse.

> **Note**: The ELT job that loads these parameters into the warehouse is configured to be initial load only. Any incremental changes to these buckets after the initial run is not captured in Oracle Utilities Analytics. Because, any changes made to the existing parameters can cause inconsistency to the loaded data. However, if a need arises to reconfigure the parameters, the parameters can be changed on the source system, and certain additional steps need to be performed to reload them into the warehouse. The star schema tables would also need to be truncated and reloaded. For the detailed steps involved in reloading the parameters, see **Reloading the Extract Parameters** on page 6-8 section.

The BI Master Configuration - Oracle Data Integrator-Based master configuration BO is delivered to contain the list of parameter values that need to be configured for use by the ELT process:

- Market Relationship Types
- Subscription Types
- Activity Category Types
- Device Event BOs to Exclude

Market Relationship Types

A service point may have several service providers (for example: distributor, retailer, etc.) where each is defined with a specific market relationship type on either the service point directly or on the service point's market indirectly. The service point-oriented facts can extract up to two of these service provider records and these two service providers are determined via the specified Market Relationship Type. If a specific service point does not have a value for the given market relationship type, the service provider is taken from the market referenced on the service point. It is not an error if no such service provider(s) is found.

Subscription Types

A service point may have several usage subscriptions. The service point-oriented facts can extract up to two of these usage subscription records and these two usage subscriptions are determined via the Subscription Type. The third subscription type, the Oracle Utilities Customer Care and Billing (CCB) Subscription Type, is used to determine the usage subscription that holds the external ID to the Customer Care and Billing service agreement so that references to the Customer Care and Billing dimensions can be populated on the facts. It is not an error if no such usage subscription(s) is found.

Activity Category Types

The Activity accumulation fact can be limited to activities of specific activity type categories. Only activities, whose Activity Type Category is specified on this master configuration BO, are included in the extract.

Device Event BOs to Exclude

The Device Event accumulation fact can be filtered to exclude certain device events. The device events, whose Device Event BO is specified on this master configuration BO, are excluded in the extract.

BI-Oriented Extendable Lookups

Every Extendable Lookup BO that was created for BI is listed in this section and it should serve as a guide/list for you when setting up the lookup values. Click the link on the **Configuration** portal to navigate to the **extendable Lookup Maintenance** portal where the lookup values can be configured. This section describes these lookups:

- Measurement Condition
- Days Since Last Normal Measurement
- Usage Snapshot Type
- Days Since Last Usage Transaction
- Unreported Usage Analysis Snapshot Type
- External System Mapping
- External System Entity Name
- External System ID Mapping

Measurement Condition

This extendable lookup is used to define anything interesting about a measurement, such as the source/type of the measurement (for example: a system estimate vs. a normal read vs. a human override). For more details on how condition codes are used on measurements, refer to the "Measurements" section in the *Oracle Utilities Meter Data Management User Guide.*

The measurement condition is a dimension on the SP Usage Snapshot star schema, which means it has dimensional attributes that can be used to slice and dice, and filter consumption.

If you want to quantify consumption by lateness and/or quality, you need to create finer grains of condition codes than those used in the base. For example, base has delivered Regular Measurement Condition as 500100. If you want to further classify a regular measurement that arrived on time, or late, or very, very late, you will have to create additional condition codes to represent them.

500101 - Regular / On Time

500102 - Regular / Late

500103 - Regular / Very Late

500104 - Regular / Very, Very Late

This means that the extendable lookup needs to be extended for additional attributes, such as lateness category, that needs to be sent to the data warehouse during extract.

Days Since Last Normal Measurement

This extendable lookup is used to define the age ranges for days since the last normal measurement was received. Each active instance in this extendable lookup is a bucket definition, where you describe what it is and what is the bucket's upper threshold. If a bucket is meant to have no upper limit (for example: 90+ days), it should be defined with an empty threshold (there should ideally only be one of this).

The lookup value codes should be defined in such a way such that when all the instances for the lookup BO are read in ascending order of the lookup value code, their corresponding upper thresholds are also in ascending order. This is important because the extract logic retrieves all the buckets in ascending order of lookup code, will compare the actual number of days on the ordered threshold, and will also apply whichever matches first (less than the upper threshold or the empty threshold). What this means is that the catch-all bucket (empty threshold) should ideally be defined so that it will be the last bucket retrieved, otherwise, any other bucket definition after the empty threshold will be of no use.

For example, the desired bucket definition is as follows: 0-30 Days, 30-45 Days, 45-60 Days and 60+ Days. This means that the following extendable lookups should be created:

Lookup Value Code	Upper Threshold	Description
10	30	0-30 Days
20	45	30-45 Days
30	60	45-60 Days
40		60+ Days

The SP Snapshot fact has a measure for the number of days since the service point's last normal measurement, and this configuration is used to find the age bucket that corresponds with that number of days since the normal measurement.

Usage Snapshot Type

This extendable lookup is used to define the granularity of the aggregated consumptions of a service point. It defines the TOU map that is applied to the service point's consumptions, where every resultant TOU and condition results in a row on the SP Usage Snapshot fact. It is also used to define the target UOM that is used to convert the source UOM prior to TOU mapping (for example: convert KW to KWH).

The Usage Snapshot Type allows an implementation to have different snapshots, such as:

• On/Off/Sh for CCF

• Day of Week for Therm

• Seasonal On/Off/Sh for Loss Adjusted kWh

This means that a given service point can have several consumption snapshots (but most implementations will have just one).

Days Since Last Usage Transaction

This extendable lookup is used to define the bucket definitions for the number of days since the last usage transaction was created for the service point. These age bucket definitions are used while extracting data for the Unreported Usage Snapshot Fact.

Unlike the Days Since Last Normal Measurement, we do not define the upper threshold here because of the possibility of differing bill cycles for different customer classes. For example, residential customers that bill quarterly vs. commercial/industrial customers that bill monthly. The thresholds are defined on the service point type configuration instead. For more details, see the Service Point Configuration section.

Unreported Usage Analysis Snapshot Type

This extendable lookup is used to define the different aging snapshots that can be taken for a service point for different types of usage subscriptions. Multiple snapshots of a single service point are allowed as implementations could have multiple systems it is sending consumption to and it may need a snapshot for each.

This allows an implementation to have different snapshots, such as:

• Oracle Utilities Customer Care and Billing

• Distribution

• …

This means that a given service point can have different Unreported Usage Analysis snapshots (for different types of usage subscriptions).

External System Mapping

This extendable lookup provides a mapping between an external system and its data source indicator. If the external system is one of Oracle Utilities edge applications, the data source indicator should be the environment id on the installation options of that external system. The data source indicator is used to uniquely identify the source of the system data.

External System Entity Name

This extendable lookup defines the entity names of entity information that can be received from external systems.

External System ID Mapping

This extendable lookup provides a mapping between the external identifiers captured in Oracle Utilities Meter Data Management and the external system where that identifier is from. Since it is possible that different sets of records for a particular entity may be synchronized from multiple external systems, it is necessary to specify the identifier type to use for each external system. The key information that is captured in this mapping is the maintenance object (which identifies the MO that stores the identifier collection), the identifier type, the external system, and the entity

name (which identifies the type of entity that uses this identifier as its external primary key). For example, the Service Point MO can contain identifiers for the Oracle Utilities Customer Care and Billing Service Point ID and Oracle Utilities Customer Care and Billing Premise ID, as well as the Oracle Utilities Operational Device Management Service Point ID.

Service Point Configuration

This section lists all the service point types in the system and indicates whether the BI configuration has been setup for each of them or not. It provides a navigation link to the service point type where the necessary configuration can be setup or modified.

The service point type allows two configurations:

* Usage Snapshot Configuration
* Unreported Usage Snapshot Configuration

Usage Snapshot Configuration

This section of the service point type defines the configuration(s) to be used to take the weekly or monthly usage snapshot(s). The configuration here is used when extracting data for the Usage Snapshot fact.

The Usage Snapshot Type defines the type of usage snapshot. Its extendable lookup definition contains the TOU map (used to map the consumption), and the target unit of measure (used if it is necessary to convert the source UOM to a target UOM prior to TOU mapping).

The UOM, TOU, and SQI are used to define the source MCs value identifier that will be TOU mapped.

A given service point type can have many usage snapshot types if there are different ways to look at the monthly consumption. This is not limited to just different TOU maps, but could also be used to create snapshots of different measured values. For example, if a measurement contains two values, actual and normally used, this can be used to create a snapshot of normal usage so it can be compared to a separate snapshot of actual usage.

If the service point type does not have at least one configuration type, service points of this type do not have their usage snapshot taken.

Unreported Usage Snapshot Configuration

This section of the service point type defines the configuration(s) to be used to take the weekly or monthly unreported usage snapshot(s). The configuration here is used while extracting data for the Unreported Usage Snapshot fact.

The Unreported Usage Analysis Snapshot Type defines the type of unreported usage snapshot.

The UOM, TOU, and SQI are used to define the source MC's value identifier that will be used to calculate the amount of unreported usage in various age buckets.

The Subscription Type is the type of subscription that the analysis will be performed on for this type of snapshot. If the service point is not linked to such a subscription, the fact will be linked to a "none" usage subscription, so analysis of consumption with no usage subscription can be performed.

The Days Since UT Buckets and their corresponding descriptions are used to categorize into different age buckets the amount of consumption that has not been billed. For example: if Bucket 1 is defined as 30, Bucket 2 is 45, and Bucket 3 is 60, any unbilled consumption that is less than or equal to 30 days old will fall into the bucket 1. Any unbilled consumption that is older than 30 days but is less than or equal to 45 days old will fall into the bucket 2. Any unbilled consumption that is older than 45 days but is less than or equal to 60 days old will fall into the bucket 3. Any unbilled consumption that is older than 60 days will fall into the bucket 4.

A separate snapshot can be taken for different subscription types, so that a given service point can have multiple snapshots for a given month/week. If the service point type does not have at least

one unreported usage configuration type, service points of this type do not have their unreported usage snapshot calculated.

BI Aggregators

The following types of groupings are used in Oracle Utilities Analytics to calculate the totals by interval for every combination of the dimensions.

- **Postal Code**: This is retrieved from the service point's address information.

- **City**: This is retrieved from the service point's address information.

- **Head-End System**: This is retrieved either from the override head-end system defined on the device or from the fallback head-end system on the device type, if there's no override on the device.

- **Usage Group**: This is retrieved either the override usage group on the usage subscription or the fallback usage group on the usage subscription type, if none on the usage subscription. The Usage Group can be set up as an optional dimension, and if so, service points without primary usage subscription can be included.

- **Market**: This is retrieved from the service point's market. The Market can be set up as an optional dimension, and if so, service points not participating in a market can be included.

- **Service Provider of Role 'X'**: If the service point participates in a deregulated market, the service provider is for the role specified. The Service Provider can be set up as an optional dimension, and if so, service points that do not participate in a market and do not have any service provider for a given role can be included.

- **Service Type**: This is retrieved from the service point's details.

- **Device Type**: This is retrieved from the device currently installed on the service point.

- **Manufacturer and Model**: This is retrieved from the device. The Manufacturer and Model can be set up as optional dimensions, and if so, devices with no manufacturer/model definition can be included.

- **Geographic Code**: This is retrieved from the service point. The Geographic Code can be set up as an optional dimension, and if so, service points without geographic information can be included.

This section covers the following:

- Aggregator Measuring Components
- Setting Up Aggregation Parameters
- Creating and Aggregating BI Aggregators
- Refreshing Materialized Views

Aggregator Measuring Components

For every combination of the dimensions listed above and for each type of aggregation there is a distinct aggregator measuring component. These measurements include the aggregated totals for their constituent measuring components.

The four types of aggregators are:

- **Measured Quantity:** In this type of aggregation, the aggregated measurements of the constituent measuring components are spread across buckets as per their measurement conditions.

- **Quality Count:** In the quality count aggregation type, a count for each interval related to a constituent measuring component is placed into one of the quality buckets.

- **Timeliness Count:** In this aggregation type, a count for each interval related to a constituent measurement component is placed into one of the late buckets.

- **Timeliness Quantity:** In this aggregation type, the aggregated measurements of the constituent measuring components are spread across late buckets as per their measurement conditions.

Setting Up Aggregation Parameters

This section describes the required and recommended aggregation parameters that should be set up while configuring the Oracle Utilities Meter Data Management application in Oracle Utilities Analytics, including:

- Master vs. Sub Aggregator
- Measurement BOs
- Timeliness Master Configuration
- Determining Initial Measurement Data Timeliness
- Value Identifiers on Aggregator MC Type (Recommended)

Master vs. Sub Aggregator

Required Configuration: Nominate one of the BI aggregation types as the master aggregator, particularly if the data analysis in BI for various aggregation types is for the same set of customers. Nominating a master aggregator makes the setup and the aggregation process easier, because the master aggregator controls how and when the aggregations are performed.

The first step is to define this type's aggregator measuring component type as the master (its MC Type BO must have an Aggregation Hierarchy Type Option of "Master"). The master aggregator measuring component type controls the aggregation parameters (horizon, lag, and cutoff time), the valid measuring component types to aggregate and its sub aggregator measuring component types.

The next step is to ensure that the measuring component BO of the master aggregator measuring component type has the appropriate algorithms plugged in on it - a BO system event Find Constituent Measuring Components algorithm (which contains the logic on how to find the constituent measuring components) and an Enter algorithm on its Aggregate state (which contains the logic on how to aggregate the measurements).

The sub aggregator measuring component types can only define the value identifiers that are applicable to it. Its aggregation parameters and valid measuring component types are inherited from the master aggregator measuring component type. Similarly, the Find Constituent MC and Aggregate algorithms are defined on the master (defining these algorithms on the sub aggregators BO will be ineffectual as they will never get triggered).

> **Note**: The usage of the each type of the BI aggregation depends on the implementation. In this case, the implementation should exclude defining this type of BI aggregation as master aggregator or in any of the sub aggregator types.

Measurement BOs

Required Configuration: Define, as appropriate, the special measurement BOs for the four types of BI aggregation (described above under Aggregator Measuring Components). These are used to differentiate the aggregated measurements from normal measurements created by initial measurement data.

These business objects should be defined on the corresponding aggregation type's measurement component type. The materialized views that were built to aggregate the individual intervals use these specific measurement BOs for performance reasons. These views are accessed directly by the BI analytics.

The measurement business objects are listed as follows:

- Measured Quantity - Measured Quantity Measurement (D2-MeasuredQuantityMsrmt)

- Quality Count - Quality Count Measurement (D2-QualityCountMsrmt)

- Timeliness Count - Timeliness Count Measurement (D2-TimelinessCountMsrmt)

- Timeliness Quantity - Timeliness Quantity Measurement (D2-TimelinessQuantityMsrmt)

Timeliness Master Configuration

Required Configuration: Add definitions for the timeliness buckets in the Timeliness Master Configuration. This is where definitions for when initial measurement data is considered late, the severity of that lateness, and also the top and bottom time range for that lateness are created.

The severity of lateness can be on time, late, very late, or missing (a measurement is considered missing if it does not exist or if it's condition is either System Estimated or No Read-Other).

In addition, this Master Configuration is where the heating and cooling degree day's factors are setup.

Determining Initial Measurement Data Timeliness

Required Configuration: Mark the initial measurement data with the number of hours that it is late. This definition is used to qualify whether or not a measurement arrived on time or late. This is done on the initial measurement data-level to allow dynamic configuration of what it means for initial measurement data (and measurements) to be late without the need to re-configure the aggregation logic. There is a measure on the initial measurement data's process data that should store the number of hours the initial measurement data is late.

The base application is delivered with an Enter algorithm that calculates the initial measurement data's timeliness as the difference between the initial measurement data's end date time and its actual creation date/time in the system (Determine initial measurement data's Timeliness D2-DET-TML). This algorithm can be plugged in on the Pending state of the initial measurement data life cycle.

Value Identifiers on Aggregator MC Type (*Recommended*)

The base Enter algorithm for aggregating BI measurements (Aggregate Measurement Counts and Quantity D2-AGG-MCQ) has allocated the measurement buckets for Measured Quantity and Quality Count. While using this base algorithm, it is recommended that the following value identifiers are set up on the appropriate aggregator measuring component type, so that you looking at the aggregated data within the Oracle Utilities Meter Data Management system can see the appropriate identifier labels for each quantity:

Measured Quantity

- Measurement Value - Measured Quantity

- Value 1 - Regular Measurement Quantity

- Value 2 - Estimated Measurement Quantity

- Value 3 - User-Edited Measurement Quantity

- Value 4 - Misc Condition 1 Measurement Quantity

- Value 5 - Misc Condition 2 Measurement Quantity

- Value 6 - MC Count (per interval)

- Value 7 - Heating Degree Days

- Value 8 - Cooling Degree Days

- Value 9 - Average Consumption

Quality Count

- Value 1 - Regular Measurement Count

- Value 2 - Estimated Measurement Count

- Value 3 - User Edited Measurement Count

- Value 4 - No Measurement/No IMD Count

- Value 5 - No Measurement/IMD Exists Count

- Value 6 - No Read Outage Count

- Value 7 - No Read Other Count

- Value 8 - Missing Count

- Value 9 - Misc 1 Count

- Value 10 - Misc 2 Count

The late buckets for Timeliness Count and Quantity are configured via the Timeliness Master Configuration. Similarly, these buckets are recommended to be reflected on the corresponding aggregator measuring component type's Value Identifiers definition.

Creating and Aggregating BI Aggregators

The BI aggregators can be created and aggregated either manually or automatically, just like other the aggregators. For more information, refer to the "Aggregation" section in the *Oracle Utilities Meter Data Management Configuration Guide*.

Refreshing Materialized Views

The two types of materialized views built for each type of BI Aggregation are explained below.

- A materialized view that contains all the BI aggregator measuring components only, where the dimensional values are flattened on the view. The pseudo-dimensions in BI are listed below.

  Measured Quantity - D2_MEASR_QTY_MV

  Quality Count - D2_QUALITY_CNT_MV

  Timeliness Count - D2_TIMELINESS_CNT_MV

  Timeliness Quantity - D2_TIMELINESS_QTY_MV

- A materialized view that contains all the aggregated measurements for BI only. The pseudo-facts in BI are listed below.

  Measured Quantity - D2_MEASR_QTY_AGR_MV

  Quality Count - D2_QUALITY_CNT_AGR_MV

  Timeliness Count - D2_TIMELINES_CNT_AGR_

  Timeliness Quantity - MVD2_TIMELINES_QTY_AGR_MV

Whenever the dimension scanning or the aggregation is done, it is important that these materialized views are refreshed, so that these materialized views contain the latest data set. An idiosyncratic batch job is delivered to perform this refresh, called Materialized View Refresh (D2-MVREF). This batch job can refresh all the materialized views in one run, so long as the materialized view names are provided as batch parameters in the run.

Setting Up the Dynamic Usage Group Extracts

The Oracle Utilities Analytics Meter Data Analytics allows for usage groups to be used for slicing and dicing the meter data analytics facts. However, in Oracle Utilities Meter Data Management, usage groups are determined dynamically via an algorithm plugged in on the usage subscription's

type. Due to this dynamic configuration of usage groups, it is difficult to determine and replicate the logic that will be involved in determining the usage group. A download table is instead created which will store the effective usage group(s) for usage subscriptions.

Extraction

An idiosyncratic batch process can be run on Initial Load or Incremental Load mode that will be responsible for determining the effective dynamic usage group of usage subscriptions and writing the results into the US usage snapshot download table. This batch job is recommended to run the incremental mode daily, to ensure that data in the download table will not be stale. Run the "Usage Subscription Usage Group Download (D2-DUGDL)" batch process to determine the effective usage group of usage subscriptions.

Setting Up Snapshot Fact Extracts

The Oracle Utilities Analytics Meter Data Analytics include four snapshot facts which capture several SP-based information:

- Consumption Snapshot

- Service Point Snapshot

- Service Point Unreported Usage Snapshot

- Service Point VEE Exception Snapshot

Taking these weekly or monthly snapshots involve massaging and/or transforming the measurement data by applying a few core Oracle Utilities Meter Data Management functionalities such as converting scalar consumptions to intervals, axis conversion and TOU mapping. The huge volume of measurement data and these core Oracle Utilities Meter Data Management functionalities cannot easily be replicated in the BI data warehouse. Hence, these snapshots will be taken in Oracle Utilities Meter Data Management via idiosyncratic batch jobs which will write the results into download tables. These download tables are the ones that the ELT process will replicate and load to the data warehouse.

Consumption Snapshot

The Consumption snapshot allows KPIs that report on every service point's measured consumption. Once a month (or once a week), the system applies a TOU map to every active service point's consumption; every resultant TOU code, quantity, and measurement condition results in a separate row on the Consumption Snapshot fact.

Setup

The following is a list of configuration steps required for a successful SP Usage Snapshot fact extract.

1. Navigate to the appropriate Service Point BO and ensure that the correct algorithm is plugged in on the system event Usage Snapshot (The system event Usage Snapshot on the Service Point BO controls if and how snapshots are taken for the service point. If there is no such algorithm plugged in on the service point's BO, the service point is skipped):

   a. Deactivate the algorithm "Aggregate SP Usage Snapshot and Write to Flat File (D2-SP-CA)" if the algorithm is plugged in and has not been deactivated yet.

   b. Plug in the algorithm "Aggregate SP Usage Snapshot and Write to Download Table (D2-SP-USG-DL)" for system event Usage Snapshot.

      Note: If the Business Object Algorithm table's caching regime is configured to be "Cached for Batch," it is necessary to clear the cache in order for the algorithm changes to take effect. This can be done by either restarting the thread pool worker or running the Flush All Cache (F1-FLUSH) batch job.

2. Setup the Usage Snapshot Type(s). Refer to the BI Oriented Extendable Lookups section under **BI Configuration** Portal for details on how to setup this data.

3. Configure the Service Point Type's usage snapshot configuration. The Service Point Type holds the below information that controls the extract:

   a. Usage Snapshot Type

   b. UOM/TOU/SQI - These are used to define the source measuring component's value identifier that will be TOU mapped. If there are multiple measuring components linked to the service point with such a combination of UOM/TOU/SQI, all will be mapped. If the service point has no measuring components with this combination, it will be skipped.

A given Service Point Type can have many Usage Snapshot Types for different ways to look at the monthly consumption. This is not limited to just TOU maps; this feature can also be used to create snapshots of different measured values.

For example: If a measurement contains two values (actual and normally used), use this feature to create a snapshot of normal consumption, so it could be compared to a different snapshot of actual consumption.

If the Service Point Type does not have at least one entry in the Usage Snapshot Configuration list, it means service points of this type do not have their snapshot taken.

Extraction

The Usage Snapshot fact has its own idiosyncratic batch process that takes the monthly or weekly consumption snapshot of all active service points in the system. Run the "Service Point Usage Snapshot Usage Download (D2-USGDL)" batch process to retrieve the service point's consumptions and populate the SP Usage Snapshot download table.

Service Point Snapshot

The Service Point Snapshot fact allows KPIs that report on the number of service points and/or devices that are installed over time. Once a month (or once a week), the system takes a snapshot of every service point in the system, extracting service point-oriented information as well as information about the service point's install event as of the snapshot date.

Setup

To configure a Service Point Snapshot fact extract successfully, follow these steps:

1. Navigate to the appropriate Service Point BO and ensure that the correct algorithm is plugged in on the system event Service Point Snapshot (The system event Service Point Snapshot on the service point business object controls if and how snapshots are taken for the service point. If there is no such algorithm plugged in on the service point's BO, the service point is skipped.)

   a. Deactivate the algorithm "Take SP Snapshot and Write to Flat File (D1-SPSNAP-SE)", if the algorithm is plugged and has not been deactivated yet.

   b. Plug in the algorithm "Take SP Snapshot and Insert to Download Table (D1-SPSNAP-DL)".

      Note: If the Business Object Algorithm table's caching regime is configured to be "Cached for Batch", it is necessary to clear the cache in order for the algorithm changes to take effect. This can be done by either restarting the thread pool worker or running the Flush All Cache (F1-FLUSH) batch job.

2. Setup the Days Since Last Normal Measurement. Refer to the BI Oriented Extendable Lookups section under **BI Configuration** portal for details on how to setup this data. This will be used to reference the dimension containing the specific age bucket that corresponds with the number of days since the last normal measurement of the service point was received.

Extraction

The Service Point Snapshot fact has its own idiosyncratic batch process that takes the monthly or weekly snapshot of all active service points in the system. Run the "Service Point Snapshot

Download (D1-SPSDL)" batch process to extract service point's information and populate the SP Snapshot download table.

Service Point Unreported Usage Snapshot

The Unreported Usage Analysis Snapshot fact allows KPIs that show consumption that has not appeared on usage transactions. Once a month (or once a week), the system reviews all active service points and determines the consumption that has taken place since the most recent usage transaction in the "sent" state.

Setup

To successfully configure the Unreported Usage Analysis Snapshot extract, follow these steps:

1. Navigate to the appropriate Service Point BO and ensure that the correct algorithm is plugged in on the system event Unreported Usage Analysis Snapshot (This system event on the service point BO controls if and how snapshots are taken for the service point. If there is no such algorithm plugged in on the service point's BO, the service point is skipped.)

   a. Deactivate the algorithm "Analyze Unreported Usage Snapshot and Write to Flat File (D2-SP-UT-AGE)", if the algorithm is plugged and has not been deactivated yet.

   b. Plug in the algorithm "Take SP Unreported Usage Snapshot and Insert to Download Table (D2-SPUT-DL)".

      Note: If the Business Object Algorithm table's caching regime is configured to be "Cached for Batch", it is necessary to clear the cache in order for the algorithm changes to take effect. This can be done by either restarting the thread pool worker or running the Flush All Cache (F1-FLUSH) batch job.

2. Set up the Days Since Last Usage Transaction. This will be used to reference the dimension containing the specific age bucket that corresponds to the number of days since the most recent "sent" usage transaction of the service point. See Unreported Usage Analysis Snapshot Type dimension for setup instructions.

3. Set up the Unreported Usage Analysis Snapshot Type. See Unreported Usage Analysis Snapshot Type dimension for setup instructions.

4. Configure the Service Point Type's unreported usage analysis snapshot configuration. The Service Point Type holds the following information that controls the extract:

   a. Unreported Usage Analysis Snapshot Type

   b. UOM/TOU/SQI - These are used to define the source measuring component's value identifier that will be used to calculate the "unbilled" consumption in the various age buckets. If there are multiple measuring components linked to the service point with such a combination of UOM/TOU/SQI, all will be mapped. If the service point has no measuring components with this combination, it will be skipped.

   c. Subscription Type - This is the type of subscription that the analysis will be performed on for this type of snapshot; if the service point is not linked to such a subscription, the fact will be linked to the "none" US, so that the analysis of consumption with no US can be performed.

   d. The Days Since UT Buckets and their corresponding descriptions - These are used to categorize the "unbilled" consumption into different age buckets. For example, if Bucket 1 is defined as 30; Bucket 2 is 45; and Bucket 3 is 60: Any unbilled consumption that is less than or equal to 30 days old will fall into bucket 1. Any unbilled consumption that is older than 30 days but is less than or equal to 45 days old will fall into bucket 2. Any unbilled consumption that is older than 45 days but is less than or equal to 60 days old will fall into bucket 3. Any unbilled consumption that is older than 60 days will fall into bucket.

A separate snapshot can be taken for different subscription types so that a given service point can have multiple snapshots for a given month/week. If the Service Point Type does not have at least

one entry in the Unreported Usage Snapshot Configuration list, it means that the service points of this type do not have their snapshot taken.

Extraction

The Unreported Usage Analysis Snapshot fact has its own idiosyncratic batch process that takes the monthly or weekly snapshot of all active service points in the system. Run the "SP Unreported Usage Snapshot Download (D2-UUSDL)" to extract service point's information and populate the SP Unreported Usage download table.

Service Point VEE Exception Snapshot

The Service Point VEE Exception Snapshot fact allows KPIs that report on VEE Exceptions. Once a month (or once a week), the system looks at every service point and count the amount of initial measurement data with and without exceptions. For initial measurement data with exceptions, it further subdivides the count by VEE Exception Type, IMD Type, VEE Severity, VEE Group, and Rule; every resultant subtotal will have a row on the VEE Exception Snapshot fact.

Setup

To successfully configure a Service Point VEE Exception Snapshot fact extract, follow these steps:

Navigate to the appropriate Service Point BO and ensure that the correct algorithm is plugged in on the system event SP VEE Exception Snapshot (This system event on the service point BO controls if and how snapshots are taken for the service point.
If there is no such algorithm plugged in on the service point's BO, the service point is skipped.)

a. Deactivate the algorithm "Aggregate SP VEE Exceptions and Write to Flat File (D2-SPVEEEXC)", if the algorithm is plugged and has not been deactivated yet.

b. Plug in the algorithm "Aggregate SP VEE Exceptions and Insert to Download Table (D2-SPVEE-DL)".

Note: If the Business Object Algorithm table's caching regime is configured to be "Cached for Batch", it is necessary to clear the cache in order for the algorithm changes to take effect. This can be done by either restarting the thread pool worker or running the Flush All Cache (F1-FLUSH) batch job.

Extraction

The Service Point VEE Exception Snapshot fact has its own idiosyncratic batch process that takes the monthly or weekly snapshot of all active service points in the system. Run the "SP VEE Exception Snapshot Download (D2-VEEDL)" to extract service point's information and populate the SP VEE Exception download table.

Configuring ELT

The Oracle Utilities Analytics product is extensible and configuration driven. There are multiple configurations that need to be done on the out of the box product. Some of these configurations are applicable system wide. And, some are applicable to the source products and some are entity and instance specific.

This section covers the configuration steps that are part of the initial deployment:

•   Oracle Utilities Analytics Basic Setup

•   Oracle Utilities Analytics Global Configurations

Oracle Utilities Analytics Basic Setup

The global configurations apply to all entities or affect the behavior of the system as a whole. There are two components to the global configuration. Oracle Utilities Analytics global configurations apply to the system and are applicable to each entity. The source product specific global configurations apply to entities associated with the specified source product. Oracle Utilities Analytics basic setup is completed only when these configurations have been set up correctly.

Oracle Utilities Analytics Global Configurations

To access the **Global Configuration** page, perform the following steps:

1. Open the browser to go to the **Oracle Utilities Analytics Administration** user interface.

2. Click on the **ETL Configurations** tab.

3. On the sidebar, click on the item **Global Configurations** as shown below.

The following screenshot shows the configuration items applicable to the entire product. The first column shows the product for which the configuration applies to.
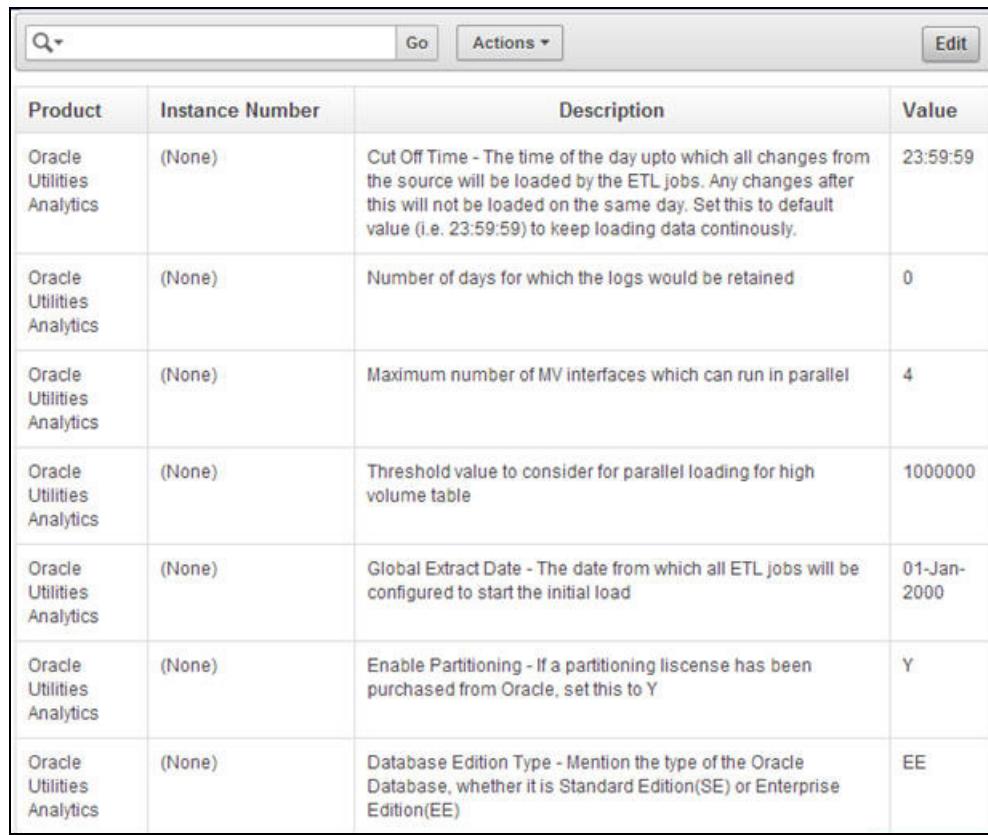
Product	Instance Number	Description	Value
Oracle Utilities Analytics	(None)	Cut Off Time - The time of the day upto which all changes from the source will be loaded by the ETL jobs. Any changes after this will not be loaded on the same day. Set this to default value (i.e. 23:59:59) to keep loading data continously.	23:59:59
Oracle Utilities Analytics	(None)	Number of days for which the logs would be retained	0
Oracle Utilities Analytics	(None)	Maximum number of MV interfaces which can run in parallel	4
Oracle Utilities Analytics	(None)	Threshold value to consider for parallel loading for high volume table	1000000
Oracle Utilities Analytics	(None)	Global Extract Date - The date from which all ETL jobs will be configured to start the initial load	01-Jan-2000
Oracle Utilities Analytics	(None)	Enable Partitioning - If a partitioning liscense has been purchased from Oracle, set this to Y	Y
Oracle Utilities Analytics	(None)	Database Edition Type - Mention the type of the Oracle Database, whether it is Standard Edition(SE) or Enterprise Edition(EE)	EE

The parameter descriptions are self explanatory. However, some additional detail needs to be mentioned about the partitioning parameter.

> **Note**: The *Partitioning* feature is an additional license on top of the Oracle Enterprise Edition. Set this property to 'Y' if you have licensed the partitioning features.
>
> Some features such as *database parallelism* is not available in the Standard Edition. Setting this property ensures that the ELT processes appropriately utilize the underlying database features for processing the loads.

Editing Global Configuration

To edit the Global Configurations, perform the following steps:

1. Login to **Oracle Utilities Analytics Administration** user interface.

2. Navigate to **ETL Configuration** > **Global Configuration**.

3. Click **Edit** to open **Maintain Global Configuration** dialog box.

4. Set the **Database Edition** to 'EE'.

5. Click **Save** to save your changes.

Initial Setup for Star Schema

Perform the following steps to populate the default key records (See the section Default Keys in Dimensions) into all the dimensions in the data warehouse:

1. Log into **Oracle Data Integrator Studio** and navigate to **Designer** > **Load Plans and Scenarios** > **Framework**.

2. Right-click the **B1_INITIAL_SETUP_PKG Version 001** and select **Run**.

3. Select **global** as context, **WLS_AGENT** as logical agent, and then click **OK**.

4. Provide the Minimum date and maximum date range to populate the CD_DATE dimension. Uncheck the option and provide the values in the option.

   Note: Check the transactional data for the minimum and maximum date range before executing this package.

5. Go to the **Operator** and view the logs for successful execution of the package.

Adding Sources

The diagram below illustrates the basic steps involved in configuring an instance of a product as a source for Oracle Utilities Analytics.
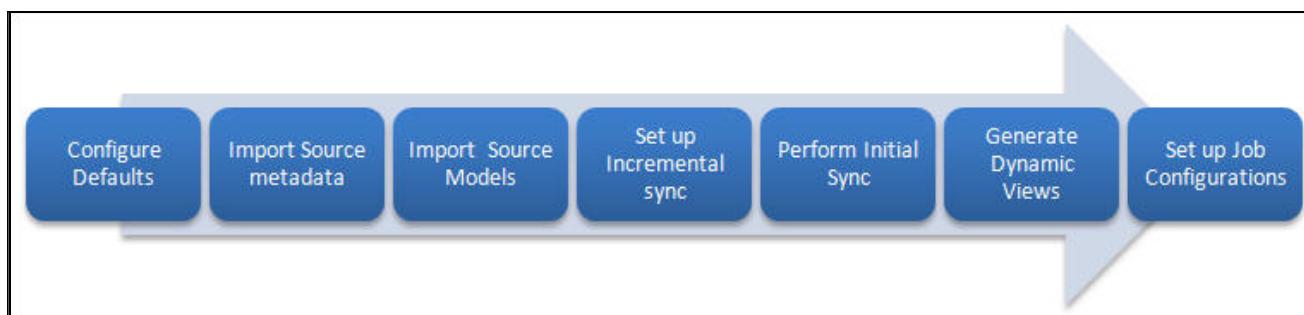
This section covers the following:

- Configuring the Socks Proxy Setup
- Invoking Source Configuration User Interface
- Using Source Configuration User Interface to Register or Upgrade Sources
  - Registering a Source
  - Upgrading a Source
- Validating Oracle GoldenGate Scripts

Configuring the Socks Proxy Setup

Follow the procedure below to configure the Socks proxy setup:

1. Login in to the source system and open the command prompt.

2. Use the **netstat** command to verify that the proxy port is free to use (*i.e.*, not active or listening for any other connections on the source system).

3. Run the following command:

```
ssh -f -N -D 0.0.0.0:<UNUSEDPORT> osusername@socks_proxy_server
```

where

- **osusername**: An account with SSH login capabilities

- **socks_proxy_server**: The hostname of the socks proxy server

4. Use the **netstat** command to verify that the port is listening.

5. When configuring the source (by running the **config.sh** GUI tool), configure SOXPROXY as *<sourcesystem>*:*<UNUSEDPORT>*.

Invoking Source Configuration User Interface

Run the following commands in the specified sequence from bin folder of the Oracle Home where Oracle Utilities Analytics was installed (This is referred to as OUA home in this document):

- **On UNIX Operating Systems:**

Add Java 8 to path variable as shown below:

```
export PATH=<Path_To_JDK_Directory>/bin:$PATH
```

```
cd bin
```

```
./config.sh
```

Example: If Oracle Home chosen during installation of Oracle Utilities Analytics is /u01/OUA/OUAProdEnv, navigate to /u01/OUA/OUAProdEnv/bin and execute:

```
cd bin
./config.sh
```

> **Important**: Note that on AIX environments, the GoldenGate configuration step of the source configuration process fails due to an existing issue with the GoldenGate Jagent. The following steps to configure the source comprise a workaround for this issue.

For an AIX operating system, perform the following steps before invoking the config.sh script:

1. Log into Oracle Data Integrator Studio.

2. Navigate to **Designer > Global Objects > Global Knowledge Modules > Journalizing (JKM)**.

3. Right-click **JKM BI Oracle to Oracle (OGG)** and select **Open**.

The Knowledge Module opens in the workspace.

4. Navigate to the **Tasks** section.

5. For the task 'Execute the apply oby file', select the **Ignore Errors** checkbox.

6. Save the KM and close it.

7. Open a new command terminal/window and navigate to OUA Home.

8. Add Java 8 to the path variable as shown below:

```
export PATH=<Path_To_JDK_Directory>/bin:$PATH
```

9. Invoke the config.sh as follows:

```
cd bin
./config.sh
```

On the successful execution of script, the GoldenGate replicat process will not be created. Perform the following steps to create the replicat process.

1. Notice that the JKM execution logs are visible in ODI Operator as B1_JRN_<source model code>.

   There will be logs available for multiple models depending on the source product being configured.

   For example: For context MWM1, the JKM logs would be available for the scenario B1_JRN_MWM1AA.

2. Right-click the execution log and click **Open**.

3. Navigate to the **Steps** section and open the log for the step 'Create Apply oby'.

4. Make a note of the steps defined in the log for dblogin, add replicat, and start replicat.

5. Open a command terminal/window and set the source ORACLE_SID and ORACLE_HOME variables.

6. Set LIBPATH as follows:

   ```
   export LIBPATH=$ORACLE_HOME/lib:$LIBPATH
   ```

7. In the Command window, change the directory to Oracle GoldenGate Home, and then execute the command below:

   ```
   ./ggsci
   ```

8. Execute the dblogin command as captured in step 4 above to log into the database.

9. After the database login, execute the add replicat and the start replicat commands captured in step 4.

10. On successful completion of these commands, execute the following command to check if the replicat process is up and running:

    ```
    info all
    ```

Repeat the steps 1 to 10 for the other models of the product.

- **On Windows Operating Systems**:

  Add Java 8 to path variable as shown below:

  ```
  set PATH= <Path_To_JDK_Directory>\bin;%PATH%
       cd bin
  config.cmd
  ```

  Example: If Oracle Home chosen during installation of Oracle Utilities Analytics is C:\OUA\OUAProdEnv, navigate to C:\OUA\OUAProdEnv\bin and execute:

  ```
  cd bin
  config.cmd
  ```

Using Source Configuration User Interface to Register or Upgrade Sources

The **Configuration Type** page appears and has three options:

- **Configure ETL**: For configuring the ETL component

- **Configure Dashboards:** For configuring the Dashboards component

- **Configure Source**: For configuring Source

Select **Configure Source** option.

The **Source Configuration Type** page launches and has two options:

• **Register Source**: The Register Source feature allows you to register a source.

• **Upgrade Source**: The Upgrade Source feature allows you to upgrade a registered source.

Registering a Source

To register a source, perform the following steps:

1. On the **Configuration Type** page, select **Register Source** option and click **Next**.

2. The **Source Product** page appears.

3. Select the source product from the **Source Product** list.

> **Note**: To register Oracle Utilities Operational Device Management (ODM) as the source product, you should register it as a Oracle Utilities Work and Asset Management (WAM) source. It ensures a smooth transition when you purchase a Oracle Utilities Work and Asset Management license in the future and setup the Oracle Utilities Work and Asset Management system on top of the existing Oracle Utilities Operational Device Management system. Registering as a Oracle Utilities Work and Asset Management source instead of Oracle Utilities Operational Device Management system does not reduce any functionality. Instead it gives the technical flexibility to start using Oracle Utilities Work and Asset Management ETL objects whenever the Oracle Utilities Work and Asset Management data is set up in the same source system.

> However, do note that the Oracle Utilities Work and Asset Management Analytics license is needed to use the Oracle Utilities Work and Asset Management ETL objects. Till then, you can only use the Oracle Utilities Operational Device Management specific objects as mentioned in the *Oracle Utilities Extractors and Schema for Oracle Utilities Operational Device Management Data Mapping Guide*.

4. Select the instance number from the **Instance Number** list and click **Next**.

5. The **Source Details** page appears. Enter the following details in the respective fields and Click **Next**:

Field Name	Description	Value
DB Host	This is the source database host name.	
DB Port	This is the source database port.	The default port is 1521
DB Service Name	This is the source database service name.	
DB Home Path	This is the source database home installed location. In case GoldenGate for source is not installed on the source database server, provide the Oracle client home location on the server on which GoldenGate is installed.	
Drill Back URL	This is the drill back URL for the source database.	
DB Schema Name	This is the source schema name.	
Extract Start Date (YYYYMMDD)	This is the date from which data should be extracted from the source.	

Field Name	Description	Value
Socks Proxy	This is the socks proxy host name and port number separated by a colon (:). For example,	Provide the value only if a socks proxy has been setup; otherwise, leave the field blank. See **Configuring the Socks Proxy Setup** on page 5-45 for details.

6. The **GoldenGate Details** page appears. Enter the following details in the respective fields and Click **Next:**

Field Name	Description	Value
Host	This is the source GoldenGate server host.	
Home Path	This is the Oracle GoldenGate installed location on the source database server.	For example: opt/local/ggs_home
Source Database Home	This is the source database home installed location.	
Manager Port	The port number on which Oracle GoldenGate Manager is running on the Oracle GoldenGate host.	The default dynamic min port is 7830. The default dynamic max port is 7880.
Encryption Algorithm	This is the algorithm configured in Oracle GoldenGate on the source server.	Provide value AES128.
Encrypt Key	This is the Encrypt Key configured in Oracle GoldenGate on the source server.	Provide the encryptkey created while setting up GoldenGate on source database server, as specified in the section *Setting up Oracle GoldenGate on the Source Database Server* of the *Installation Guide*.
Shared Secret	This is the shared secret key configured in Oracle GoldenGate on the source server.	Refer to the instructions in *Generating the Shared Secret Password* section of the *Installation Guide* for getting this value.
GoldenGate Owner User	This is the user name of GoldenGate Owner user.	
GoldenGate Owner Password	This is the password of GoldenGate Owner user.	

7. The **Source JAgent Details** page appears. Enter the following details in the respective fields and click Next

Field Name	Description	Value
JAgent Host	The host of the Oracle GoldenGate JAgent.	
JAgent GoldenGate	This is the Oracle GoldenGate installed location where GoldenGate JAgent is running.	Example: /opt/local/ ggs_12.1.2.1.0
JAgent Port	The port number on which Oracle GoldenGate JAgent is running on the GoldenGate host.	
JAgent User	This is the JAgent user name.	
JAgent Wallet Password	This is the JAgent Wallet password.	
Confirm JAgent Wallet Password	Re enter JAgent Wallet password to confirm.	

8. The **Configuration Summary** page opens showing the log file location. Click **Configure**.

9. The **Configuration Progress** page opens showing the status. Click **Next**.

10. The **Completion Summary** page appears showing the log file location. Click **Finish.**

11. Upon completion, the status of source registration is shown in a prompt.
The detailed logs of the operation are available in the logs/system/deployodi.log file in the Oracle Utilities Analytics home.

Upgrading a Source

To upgrade a source, perform the following steps:

1. On the **Configuration Type** page, select **Upgrade Source** option and click **Next**.

2. The **Source Product** page appears. All the source contexts that are already registered will show up in the **Source Product** page.

3. Select the source product from the **Source Product** list. and click **Next**:

4. The **Source Details** page appears. The previously configured values for the selected source are visible in this page.

5. Modify the values in the page where required and Click **Next.** The following is a brief description of the fields in this page:

Field Name	Description	Value
DB Host	This is the source database host name.	
DB Port	This is the source database port.	The default port is 1521
DB Service Name	This is the source database service name.	
DB Home Path	This is the source database home installed location.	

Field Name	Description	Value
Drill Back URL	This is the drill back URL for the source database.	
DB Schema Name	This is the source schema name.	
Extract Start Date (YYYYMMDD)	This is the date from which data should be extracted from the source.	
Socks Proxy	This is the socks proxy host and port separated by a :	Provide the value only if a socks proxy has been setup. Leave the field blank otherwise.

> **Note**: While upgrading a source that was registered using Oracle Utilities Analytics versions prior to Oracle Utilities Analytics 2.6, values for the database schema name and drillback URL will not appear by default. These parameters must be entered to proceed with upgrade of the source.

6. The **GoldenGate Details** page appears. The previously configured values for the selected source are visible in this page.

7. Modify the values where required and Click **Next.** The following is a brief description of the fields in this page:

Field Name	Description	Value
Host	This is the source GoldenGate server host.	
Home Path	This is the Oracle GoldenGate installed location on the source database server.	For example: opt/local/ggs_home
Source Database Home	This is the source database home installed location.	
Manager Port	The port number on which Oracle GoldenGate Manager is running on the Oracle GoldenGate host.	The default dynamic min port is 7830. The default dynamic max port is 7880.
Encryption Algorithm	This is the algorithm configured in Oracle GoldenGate on the source server.	Provide value AES128.
Encrypt Key	This is the Encrypt Key configured in Oracle GoldenGate on the source server.	Provide the encryptkey created while setting up GoldenGate on source database server, as specified in the section *Setting up Oracle GoldenGate on the Source Database Server* of the *Installation Guide.*
Shared Secret	This is the shared secret key configured in Oracle GoldenGate on the source server.	Refer to the instructions in *Generating the Shared Secret Password* section of the *Installation Guide* for getting this value.

Field Name	Description	Value
GoldenGate Owner User	This is the user name of GoldenGate Owner user.	
GoldenGate Owner Password	This is the password of GoldenGate Owner user.	

> **Note**: While upgrading a source that was registered in using Oracle Utilities Analytics versions prior to Oracle Utilities Analytics 2.6, the parameter values for Oracle GoldenGate Owner User and Oracle GoldenGate Owner Password are not populated by default. These parameters must be re-entered to proceed with upgrade of the source.

8. The **Source JAgent Details** page appears. Enter the following details in the respective fields and click **Next**

Field Name	Description	Value
JAgent Host	The host of the Oracle GoldenGate JAgent.	
JAgent GoldenGate	This is the Oracle GoldenGate installed location where GoldenGate JAgent is running.	Example: /opt/local/ggs_12.1.2.1.0
JAgent Port	The port number on which Oracle GoldenGate JAgent is running on the GoldenGate host.	
JAgent User	This is the JAgent user name.	
JAgent Wallet Password	This is the JAgent Wallet password.	
Confirm JAgent Wallet Password	Re enter JAgent Wallet password to confirm.	

9. The **Configuration Summary** page opens showing the log file location. Click **Configure**.

10. The **Configuration Progress** page opens showing the status. Click **Next**.

11. The **Completion Summary** page appears showing the log file location. Click **Finish.**

12. Upon completion, the status of source registration is shown in a prompt.
    The detailed logs of the operation will be available in the logs/system/deployodi.log file in the Oracle Utilities Analytics home.

Validating Oracle GoldenGate Scripts

The Oracle GoldenGate scripts are generated and deployed in the source and target server. Login to the Oracle GoldenGate home and check if the Extracts and replicates are in **Running** status. If the Oracle Golden gate process are not in **Running** status then start the Oracle GoldenGate process. The Initial load to the replication is done using the B1_SYNC_CONTEXT scenario from Oracle Data Integrator.

Initial Load to Replication

For initial Load from Source to Replication the B1_SYNC_CONTEXT scenario is executed after logging into Oracle Data Integrator. B1_SYNC_CONTEXT uses the database data pump functionality.

The data can be copied to replication layer using two methods:
- DB link

- Export and Import

DBLink Method: - The Sync method has to be updated in the **Admin** tool in **Global Configuration** page for the product and instance that is being configured. By default, the DBLINK method is used.

Export and Import: - The Sync method has to be updated in the **Admin** tool in **Global Configuration** page for the product and instance that is being configured. First the parameter value has to be updated to **DPUMP_EXPORT** to export the data to the file in the preconfigured oracle directory B1_DATA_DUMP_DIR. Execute the B1_SYNC_CONTEXT for the respective context.

> **Note**: The export code uses Compression feature of database to compress the export data. This requires additional license. If export and import functionality is used make sure the appropriate license are available.

Once the Data is exported move the files to the target server in the location pointing to B1_DATA_DUMP_DIR directory. Change the parameter value to **DPUMP_IMPORT** in Admin tool's **Global Configuration** page. After the parameter value is changed and the files are moved to the target server run the B1_SYNC_CONTEXT for the configured source product instance's context.

Follow the steps below to execute the B1_SYNC_CONTEXT.
1. Login to the ODI console

2. Navigate to the **Load Plans and Scenario** folder in the **Designer** tab.

3. Expand the **Framework** folder.

4. Right click on **B1_SYNC_CONTEXT Version 001** and click **Run**.

5. A popup windows opens to enter the **Context** and **Logical Agent**.
   In the example provided we have to choose context code CCB1 and Logical agent as WLS Agent.

Alternatively you can also run the B1_SYNC_CONTEXT using the ODI console by following these steps:

1. Login to the ODI console. The ODI console is deployed when the WebLogic agent for the ODI is created. The URL for ODI console would be

   ```
   http://<Weblogic Host>:<Managed Server port>/odiconsole
   ```
2. Login to Work repository using the SUPERVISOR credentials.

3. Click on the browser.

4. Navigate to the **Framework** folder using the path below
   Runtime >> Scenario/Load Plan >> Folders>> Framework

5. Right click on **B1_SYNC_CONTEXT - 001** and click **Execute**.

Upgrading Oracle Utilities Analytics

If you are upgrading Oracle Utilities Analytics and using one of these source products - Oracle Utilities Meter Data Management, Oracle Utilities Mobile Workforce Management, Oracle Utilities Network Management System whose ELT was earlier based on Oracle Warehouse Builder, certain additional steps are needed. After the step to execute the Oracle Data Integrator scenario B1_SYNC_CONTEXT for the initial data load into the replication layer has been completed successfully, note the timestamp using the query given below on the target database:

```
alter session set nls_datE_format ='DD-MON-YYYY HH24:MI:SS';
select max(COMPLETED_ON) from <Context code>REP.CDC_SYNC_LOG;
```

where <Context code> is the context Code with which the source product instance was configured like MDM1, MWM1 etc.

This timestamp is important since the Oracle Data Integrator based ELT jobs should start loading data only after timestamp. Data prior to this timestamp would have already been loaded via the earlier Oracle Warehouse Builder based ETL processes. The next step is to update the configuration of the ELT jobs with this timestamp.

For the ELT jobs of Oracle Utilities Network Management System:

```
update MDADM.B1_JOB_CONFIG
set SLICE_START_DTTM = <timestamp_noted_earlier>
where PROD_FLG = 'NMS'
and TARGET_ENTITY_ID in (
select TARGET_ENTITY_ID from MDADM.B1_TARGET_ENTITY
where ENTITY_NAME in
('CD_ACCT', 'CD_ADDR', 'CD_CALL_INFO', 'CD_CITY', 'CD_CREW',
'CD_CTRL_ZONE', 'CD_DEVICE', 'CD_EVENT', 'CD_EVENT_STATUS',
'CD_FEEDER', 'CD_METER', 'CD_PER', 'CD_PHASE', 'CD_PREM', 'CD_SNL',
'CD_STORM','CD_STORM_OUTAGE_TYPE','CD_SW_PLAN','CD_SW_PLAN_STATE',
'CD_USER', 'CF_CUST_RECENT_OUTG', 'CF_CUST_RST_OUTG',
'CF_RECENT_CALL','CF_RECENT_CREW', 'CF_RECENT_JOB', 'CF_RST_CALL',
'CF_RST_CREW', 'CF_RST_JOB', 'CF_SW_PLAN', 'CF_SW_PLAN_STATE'));
```

For the snapshot fact ETL jobs alone, the timestamp should be updated for the first second of the next month. For example, if the last sync timestamp noted earlier is "12-NOV-2013 04:02:30", then the arrears ETL job should have the timestamp as "01-DEC-2013 00:00:00". Use the below query on the target database to update the timestamp on the ETL jobs. Derive the timestamp to be used based on the timestamp noted in the earlier.

```
update MDADM.B1_JOB_CONFIG
set SLICE_START_DTTM = <First second of next month to load>
where PROD_FLG = 'NMS'
and TARGET_ENTITY_ID in
(select TARGET_ENTITY_ID from MDADM.B1_TARGET_ENTITY
where ENTITY_NAME in
('CF_FEEDER_DLVRD_LOAD','CF_OUTG',
'CF_CTRL_ZONE_OUTG','CF_CITY_OUTG'));
```

For the ETL jobs of Oracle Utilities Meter Data Management:

```
update MDADM.B1_JOB_CONFIG
set SLICE_START_DTTM = <timestamp_noted_earlier>
where PROD_FLG = 'MDM'
and TARGET_ENTITY_ID in (
select TARGET_ENTITY_ID from MDADM.B1_TARGET_ENTITY
where ENTITY_NAME in
('CD_ADDR','CD_CONS_TYPE','CD_CONTACT','CD_DAYS_LASTUT_TYPE',
'CD_DAYS_LAST_MSRMT','CD_DEVICE_ACTIVITY_STATUS',
'CD_DEVICE_ACTIVITY_TYPE','CD_DEVICE_EVT_STATUS',
'CD_DEVICE_EVT_TYPE','CD_EXCP_SEV','CD_EXCP_TYPE',
'CD_IE_STATUS','CD_IMD_TYPE','CD_MC','CD_MSRMT_COND',
```

```
'CD_MTR_DEVICE','CD_SP','CD_SPR','CD_SP_STATUS',
'CD_SP_UT_AGE_TYPE','CD_UOM_TOU','CD_UOM_TOU_SQI',
'CD_US','CD_USAGE_GROUP','CD_VEE_RULE',
'CF_DEVICE_EVT','CF_DEVICE_ACTIVITY','CF_INSTALL_EVT','CF_SP'));
```

For the Snapshot fact ETL jobs alone, the timestamp should be updated for the first second of the next month. For example, if the last sync timestamp noted earlier is "12-NOV-2013 04:02:30", then the arrears ETL job should have the timestamp as "01-DEC-2013 00:00:00". Use the below mentioned query on the target database to update the timestamp on the ETL jobs. Derive the timestamp to be used based on the timestamp noted in the earlier.

```
update MDADM.B1_JOB_CONFIG
set SLICE_START_DTTM = <First second of next month to load>
where PROD_FLG = 'MDM'
and TARGET_ENTITY_ID in
(select TARGET_ENTITY_ID from MDADM.B1_TARGET_ENTITY
where ENTITY_NAME in
('CF_CONSUMPTION','CF_SP_SNAP','CF_SP_UT_AGE','CF_VEE_EXCP'));
```

For the ETL jobs of Oracle Utilities Mobile Workforce Management:

```
update MDADM.B1_JOB_CONFIG
set SLICE_START_DTTM = <timestamp_noted_earlier>
where PROD_FLG = 'MDM'
and TARGET_ENTITY_ID in (
select TARGET_ENTITY_ID from MDADM.B1_TARGET_ENTITY
where ENTITY_NAME in
('CD_ADDR','CD_CREW','CD_APPT_TM','CD_APPT_TM_OF_DAY',
 'CD_CREW_SHIFT','CD_CREW_TM_USG','CD_EARLY_LOGOFF_TM',
 'CD_LATE_LOGON_TM','CD_RESP_TM_DEV','CD_SERVICE_AREA',
 'CD_SHIFT_BO_STATUS','CD_TASK_BO_STATUS','CD_TASK_TYPE',
 'CD_TRAVEL_DIST_DEV','CD_TRAVEL_DUR_DEV','CD_WORK_DUR_DEV',
 'CF_CMP_SHIFT','CF_CREW_TASK','CF_FLD_ACTIVITY'));
```

Initiating the ELT Jobs

To start the ELT jobs, two things need to be done. The first task is to setup a scheduler in Oracle Data Integrator which would run continuously looking for jobs to execute. The next task is to enable the specific set of ELT jobs to actually load the target star schemas.

Setting up the Scheduler

Perform the following steps:

1. In **Oracle Data Integrator Studio**, navigate to **Designer** > **Load Plans and Scenarios** > **Framework** > **B1_RUN_ALL Version 001** > **Scheduling**.

2. Right-click scheduling and select **New Scheduling** option.

3. Select the source context as global, agent as **WLS_AGENT** and the log level as 1.

4. To specify how often the scheduler should run, navigate to the **Execution Cycle** tab of the Scheduler and select the '**Many Times**' option. Set the interval between repetitions.

5. Navigate to **Topology** > **Agents** > **OracleDIAgent**.

6. Right-click on **OracleDIAgent** and click **Update Schedule**.

Enabling the ELT Jobs

Using the Oracle Utilities Analytics Administration tool, navigate to **ETL Configuration** > **Job Configuration.** The jobs can be enabled individually one by one by updating each record in this page. To mass enable multiple jobs at the same time, use the **Enable Jobs** button on the top right side of the page. It is recommend that customers enable the jobs in a progressive manner such as enabling the dimensions first, the facts next and finally the materialized view jobs. This helps reduce the load on the server during the initial data load.

Alternately for initial load the Oracle data integrator scenario B1_DWH_LOAD can be trigger for the given context to load the data. This scenario would enable the dimensions initially. The facts would be triggered once the dependent Dimension load is completed. The Materialized view are enabled only when the Dimensions and Facts initial load is completed. Follow the below steps to start the B1_DWH_LOAD.

1. Login to ODI client.

2. Navigate to the **Load Plans and Scenario** folder.

3. Navigate to the **Framework** folder.

4. Right-click **B1_DWH_LOAD**.

5. Select **Run** and provide the context. Then, select the **Agent** as **WLSAGENT**.

Configuring the Security

The security scripts have to be executed after the initial load and the Materialized views are created for the given source product instance. For configuring the security, perform the following steps:

1. In the Oracle Utilities Analytics database component installation package, navigate to the ..\BI270\DWADM folder

2. Run OraGenSec.

   In **Windows** environment:

   a. Add Java 8 to path variable as shown in example below:

   ```
   set PATH= C:\Java\jdk1.8.0_121\bin;%PATH%
   ```
   b. Set classpath pointing to OraDBI.jar and all dependency jars.

   ```
   set CLASSPATH=<Path to BI270>\BI270\DWADM\Jarfiles\*
   ```
   c. Run OraGenSec as shown below:

   ```
   java com.oracle.ouaf.oem.install.OraGenSec -l oragensec.log -d
   DWADM,DWADM,jdbc:oracle:thin:@<DATABASE SERVER>:<DATABASE
   PORT>/<ORACLE SERVICE NAME> -a A -r DW_READ,DW_USER -u
   DWREAD,DWUSER
   ```

   In **Unix** environment:

   a. Add Java 8 to path variable as shown in example below:

   ```
   export PATH=/scratch/softwares/jdk1.8.0_121/bin:$PATH
   ```
   b. Set classpath pointing to OraDBI.jar and all dependency jars.

   ```
   export CLASSPATH=<Path to BI270>/BI270/DWADM/Jarfiles/*
   ```
   c. Run OraGenSec as shown below:

   ```
   java com.oracle.ouaf.oem.install.OraGenSec -l oragensec.log -d
   DWADM,DWADM,jdbc:oracle:thin:@<DATABASE SERVER>:<DATABASE
   PORT>/<ORACLE SERVICE NAME> -a A -r DW_READ,DW_USER -u
   DWREAD,DWUSER
   ```

Configuring the Analytics

This section describes the various Administration Dashboards and the details about configuration of dashboards.

- Maintaining the Administration Dashboards
- Configuring the Dashboards
- Deploying the Custom MapViewer

Maintaining the Administration Dashboards

The following dashboard pages are described in this section.

- Maintaining the Base Fields
- Maintaining the Custom Fields

Maintaining the Base Fields

This dashboard allows you to provide an override description for the fields delivered along with the product. These fields contain descriptions, which appear on the report titles and the column titles for the dashboards delivered with the product.

> **Note**: For the complete details on the field labels, refer to the Configuring the Labels in the section Configuring the Dashboards.

Maintaining the Custom Fields

This dashboard allows you to define additional fields and your description, which you may want to add as a part of customization.

This dashboard has the following pages to allow you to add, edit, and delete your own fields:

- Update
- Insert
- Delete

> **Note**: For the complete details on the field labels, refer to the Configuring the Labels in the section Configuring the Dashboards.

Configuring the Dashboards

This section describes the configuration dashboards in Oracle Utilities Analytics, including:

- Configuring the Oracle Utilities Meter Data Management Answers
- Configuring the Labels
- Configuring the Spatial Data
- Viewing the About Page

Configuring the Oracle Utilities Meter Data Management Answers

The following Oracle Utilities Meter Data Management Answers should be configured before viewing the data:

- Tamper Events Answer (**Overview** dashboard)
- Usage Unreported for > 30 Days (**Overview** dashboard)
- Percent of Normal Intervals (**Overview** dashboard)

- Percent of On-Time Intervals (**Overview** dashboard)

- Degree Days (**Overview** dashboard)

After customizing the answers, save the reports in a separate CM catalog.

> **Note**: For details, refer to *Oracle Utilities Analytics Dashboards for Oracle Utilities Meter Data Analytics Metric Reference Guide.*

Configuring the Labels

This section describes how to create and customize the labels that appear in answers and dashboards.

> **Note**: You must have a full license of Oracle Utilities Analytics to use this feature.

This section includes the following topics:

- Overview of Labels

- Overriding the Base Labels

- Supporting the Multiple Languages

Overview of Labels

Oracle Utilities Analytics uses labels for the columns and the tables in the delivered Oracle Business Intelligence Enterprise Edition repository file when displaying the columns in the **Presentation** folders for you. These labels are displayed in the answers for the columns on the dashboards. In addition, the answers are also titled based on labels stored in the metadata displayed in report titles, sub-titles, and other dashboard strings.

The application uses labels instead of the hard coding the text values in the answers and the RPD file for supporting translation of the dashboards into the different languages and allowing easy overriding of the labels for you if you wish to customize the field label.

For example, within an answer, the labels can be referred to by Oracle Business Intelligence Enterprise Edition server variables. For example, the **Device Activity - Distributions** report uses this variable in the title section of the answer:

```
@{biServer.variables['NQ_SESSION.B1_RPT_DEV_ACTI_DISTRIBUTION']}
```

The **B1_RPT_DEV_ACTI_DISTRIBUTION** label is defined in the **B1_MD_FLD** table in **DWADM** schema.

For the columns in the fact and dimension tables, the labels exist for every field. For example, the **UDF1_DESCR** column in the **CD_ACCT** table has the description of the **Customer Class**, and the **Customer Class** label is displayed in the **Presentation** folder for this field.

Overriding the Base Labels

There are several reasons that an implementer may want to update an existing label:

- A field may contain data that does not match the default extracted data for that field.
  In the **CD_ACCT** example, described in the above section, you may elect to store information other than customer class in the **UDF1_DESCR** field. If an extract change is made to the default **CD_ACCT** extract, then an implementation change in the label for the **UDF1_DESCR** field of the **CD_ACCT** table at one place changes the label in all the dashboards and answers that display that field. This reason also applies if data is extracted to a User Defined Field (UDF) field that is not already having a default population.

- Even if you use the default extract code, you may choose to use some other name for the extracted data other than the default name.
  In the **CD_ACCT** example, if you execute the field extracted into the **UDF1_DESCR** field account class instead of customer class, you can make this change at one place and have it updated on all dashboards and answers.

- You may want to provide multilingual labels for your users. Oracle Utilities Analytics application provides the labels to a user based on the language selected when logging into Oracle Business Intelligence Enterprise Edition, assuming that the language is present in the **B1_MD_FLD** table. An implementation can add its own translated fields, or can download supported language packs from the Oracle Software Delivery Cloud.

  Note: The multilingual support is only provided for labels and not for the underlying data in the data warehouse. The data displayed in all the database tables is not translatable from the extract language.

Supporting the Multiple Languages

Oracle Utilities Analytics is released with default support for English labels on all the dashboards and answers. Both Oracle Business Intelligence Enterprise Edition and Oracle Utilities Analytics support the multiple languages.

The default language on the **Login** page is English. However, you can select any of the supported language on the **Login** page or can change the preferred language under the **Administration** menu to view dashboards in a different language. If you have not purchased and applied the specific language pack and if you select a language other than English, the default Oracle Business Intelligence Enterprise Edition labels is still be translated in the selected language, but Oracle Utilities Analytics product specific labels appear in English.

Oracle Utilities Analytics may release various language packs depending on user demands. Hence, for the language that is already released, installing the language pack is sufficient for creating the labels needed by the dashboards.

To view the list of Oracle Utilities Analytics language pack applied on an environment, you can navigate to the **About Oracle Utilities Analytics** dashboard under the Viewing the About Page in the Oracle Business Intelligence Enterprise Edition dashboards menu.

Contact your Oracle support representative to purchase an Oracle Utilities Analytics language pack for additional language support.

To update a label for a base field, use the **Base Field Maintenance** dashboard in the **Administration** portal.

- **Table Labels**: For records that have the **Table Name** field populated, but not the **Field Name,** this label is shown in the **Presentation** folder for the fields available in this table. For example, the **CD_ACCT** table has the label '**Account Dimension**' displayed in the **Presentation** folder wherever it is used.

- **Field Labels**: For records that have both the **Table Name** and **Field Name** fields populated, this label is shown in the **Presentation** folder and on answers whenever that field is used. For example, the **UDF1_DESCR** field in the **CD_ACCT** table has the label '**Customer Class**' displayed whenever it is used in an answer or when you select it from the **Presentation** folder when creating a new answer.

- **Report Labels**: Records that have a field name, such as 'B1_RPT%' and no table name value are used for the titles of answers in the dashboards. For example, the **B1_RPT_DEV_ACTI_DISTRIBUTION** label is defined to be '**Device Activity Distribution**', and this is displayed on the **Oracle Utilities Meter Data Management** dashboard when the answer is displayed.

- **Other Labels**: All other non-report labels that have a field name, but no table name is used for calculations that are computed in the RPD logical layer for display on answers. For example, the **B1_APPT_IND_SUM** label is defined to be '**Number of Appointments**', and is used in Oracle Utilities Mobile Workforce Management answers that compute the number of crew appointments based on the **Appointment Indicator** field in the **CF_CREW_TASK** fact table.

If a base field label should be changed, then the implementation team can query the required record on the **Base Field Maintenance** dashboard; populate a value in the **Override**

Description field, and click **Update**. Once populated, the Oracle Business Intelligence Enterprise Edition Server must be restarted for the changes to take effect.

Configuring the Spatial Data

This section describes how to configure mapping for Oracle Utilities Analytics. It includes the following topics:

- Loading the Geographical Data
- Setting Up the Network Model Spatial Data in Outage Analytics
- Configuring the Google Map Tile Layer
- Implementing the Maps

Loading the Geographical Data

In order to place information on a geographic map, data in the data warehouse must match geographic data (themes) that are configured in Oracle MapViewer.

The standard map answers delivered with Oracle Utilities Analytics include maps that query state, city, county, postal codes, and network model summary data. As Oracle Utilities Analytics does not have access to this spatial data (and as each user may require different spatial data), you must set up the geographic themes used in the maps.

> **Note**: For details regarding setting up these standard spatial themes, refer to *Oracle Utilities Analytics Installation Guide.*

The installation instructions refer to shape files downloaded from the US Census Bureau. However, the shape files can also be used for the state, city, county, and zip code boundaries. The only requirement is that the names of the geographic entities in the shape file should match the corresponding name in the **CD_ADDR** table. This is not usually a problem for postal code data, but can be an issue for city and county names, as different sources may use different names to refer to the geographic places. Make sure that after loading the **MapViewer shapefiles** that the names in the geographic tables match the names that are in the **CD_ADDR** table. If these do not match, then the maps may not display correctly.

Setting Up the Network Model Spatial Data in Outage Analytics

This section provides Oracle Utilities Network Management System (NMS) network model representation in Outage Analytics.

Oracle Utilities Network Management System provides model build process to generate geometry data for various components of network model. The geographic data is stored in the DIAGRAM_OBJECTS table.

> **Note**: For details regarding the steps required to set up the Network Management System Model Build process, refer to *Oracle Utilities Network Management System Installation Guide.*

To plot these elements in Outage Analytics Dashboards, the DIAGRAM_OBJECTS table should be replicated as per the instructions available in the section **Installing US State Spatial Data** in *Oracle Utilities Analytics Install Guide.*

The proper mechanism should be in place to be up-to-date with the DIAGRAM_OBJECTS table in the Source database.

To ensure that the network model can be displayed without coordinate translations during runtime, one of the geometry columns should use the same projection coordinate system as the base map used by the Outage Analytics Outage Maps. Out of the box Oracle eLocation is used for the base map and the SRID (Spatial Reference System Identifier) is 54004. Note that the SRID is a unique value used to identify the coordinate system used in a Geographic Information System (GIS) application.

Configuring the Google Map Tile Layer

Out-of-the-box Oracle Utilities Analytics is configured to fetch the map tiles from the eLocation. However, you can switch to Google as an alternative source for the map tiles.

Perform the following steps:

1. Log onto the **MapViewer** console.
   Sample URL:
   http://<host-name>:<port>/mapviewer

2. Click the **Admin** and enter the credentials.

3. On the **Management** tab, click the **Manage Map Tile Layers**.

4. Create a new tile layer by selecting **Google**.
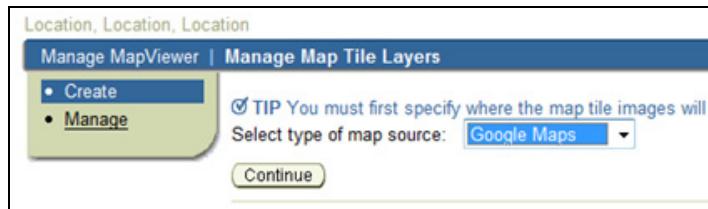
5. Enter the key fetched from Google in the **Key** field, edit the default **lib_url** to include the key value and choose the appropriate data source.
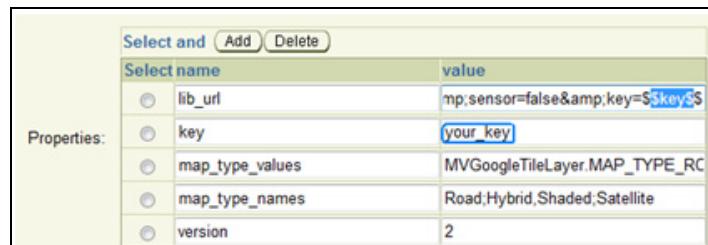
6. Click **Submit** to save the information.

7. Click **View map/Manage tiles** to verify the new tile layer shows up properly.

8. Click **Show Map** to view the map.

9. Logon to Oracle Business Intelligence Enterprise Edition Analytics and navigate to **Administration > Manage Map Data** and click on the **Background Maps** tab.
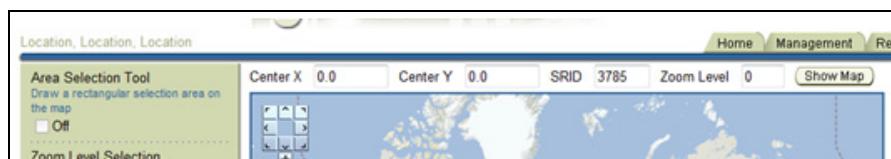
10. Import the tile layer **'GOOGLE_MAP'** created on the **MapViewer** console.

11. Add layers to the map and save it.
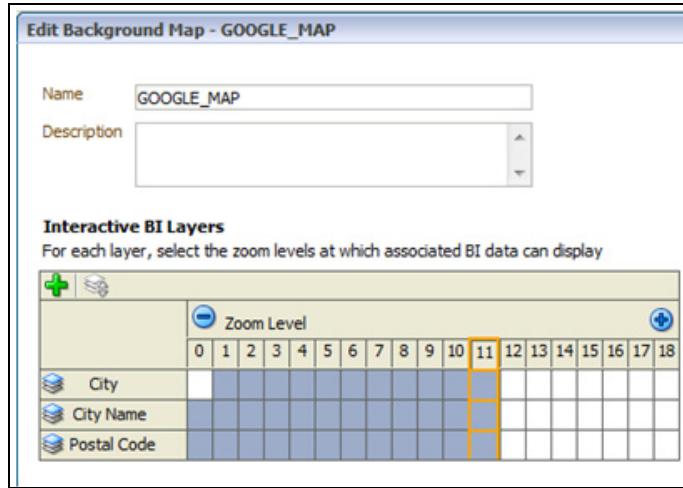
12. Subject areas will now be associated to the map.

13. Create a new answer and add a new map-view to have the data displayed on the map.
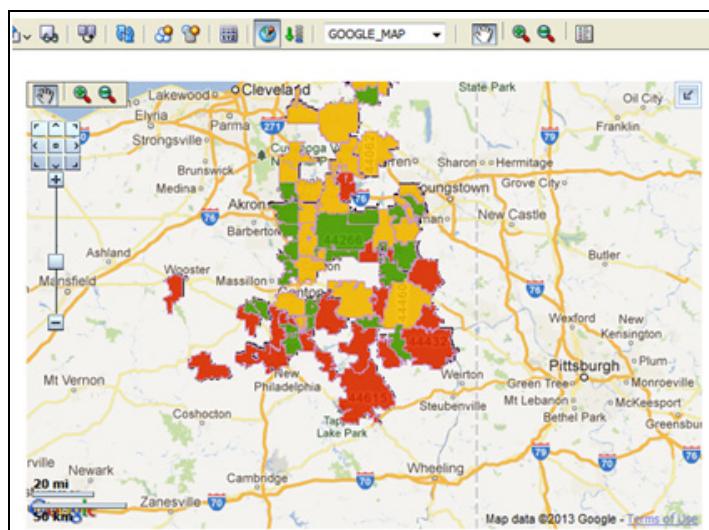
Implementing the Maps

This section describes the method of implementing maps in Oracle Utilities Analytics. The implementation method is the default implementation method for Oracle Business Intelligence Enterprise Edition 11g. This form of map can be seen in various dashboards, such as Oracle Utilities Meter Data Analytics, Oracle Utilities Mobile Workforce Analytics, and Oracle Utilities Customer Care and Billing Analytics. Using this method, you can create new answers using the MapView. This view uses the configuration defined under the **Administration** menu in the **Manage Map Data**.

The layers, background maps, and the images being used in the map must be defined in this page. The key column and geographical columns are mapped for each subject area used in the analysis. This is a one-time setup unless new subject areas are added.

> **Note**: You should not customize the map metadata until you import the Spatial Catalog file.

For customizations that involve map analysis, all the modifications must be done in a separate folder in order for those modifications to be preserved when upgrading Oracle Utilities Analytics.

Viewing the About Page

The **About** page shows information about the product name along with the current release version and patch number. It also lists all of the languages that are currently installed in the product.

Enabling Weather Information

Oracle Utilities Analytics product supports weather information display on some of its Oracle Business Intelligence Enterprise Edition map reports. Currently, this feature is limited only to the users of Outage Analytics. To enable this feature, perform these steps:

• Deploying the Custom MapViewer

• Subscribing to the Weather Service Provider

• Configuring the Weather API Details

Deploying the Custom MapViewer

This is an optional step in the installation process. Outage Analytics users who wish to enable the weather features should carry out all the steps, documented in the section **Configuring and Deploying the MapViewer** under the chapter **Oracle Utilities Analytics Initial Installation** in *Oracle Utilities Analytics Installation Guide.*

Subscribing to the Weather Service Provider

Weather information is not captured and stored anywhere in the Oracle Utilities Analytics product. This is a complex process and is beyond the scope of Oracle Utilities Analytics. There are third party vendors who provide weather information. Each vendor has their own set of APIs, which provide weather information in varied formats.

The previous step is to deploy the Oracle Utilities Analytics code, which would contact one of such weather vendors, process the data and render it on the maps. Out-of-box the Oracle Utilities Analytics code will work with Pulse APIs provided by Earth Networks. You need to subscribe these from their website (http://weather.weatherbug.com/pulseapi.html) and opt for one of their package.

> **Note**: This is a third party vendor and you need to purchase the license separately from them.

Upon successful registration and purchase, you would be provided with a consumer key and secret key. Using this, you should generate an access token using the URL mentioned below:

https://thepulseapi.earthnetworks.com/oauth20/
token?client_id=<insert_consumer_key_here>&client_secret=<insert_secret_key_here>&grant
_type=client_credentials

This should provide a JSON response with the access token as shown below:

Response format: {"OAuth20":{"access_token":{"token":"<access token>
","refresh_token":"<refresh token>","token_type":"bearer","expires_in":86399}}}

Configuring the Weather API Details

The final step is to configure couple of values for the Oracle Utilities Analytics product to make use of. More specifically, the weather API provider's URL domain name and the access token should be configured.

Log onto the Oracle Business Intelligence Enterprise Edition dashboard environment, navigate to **Administration dashboards** > **Configuration** > **Map Profile** page. Update the two records as highlighted below. Use the access token that was generated in the earlier step.

Click **Update** to save the changes. The weather configuration is now complete. To see the weather information, navigate to Outage Analytics, go to **Current Outages** dashboard > **Overlay Map** page, you would see something similar to the below snapshot.
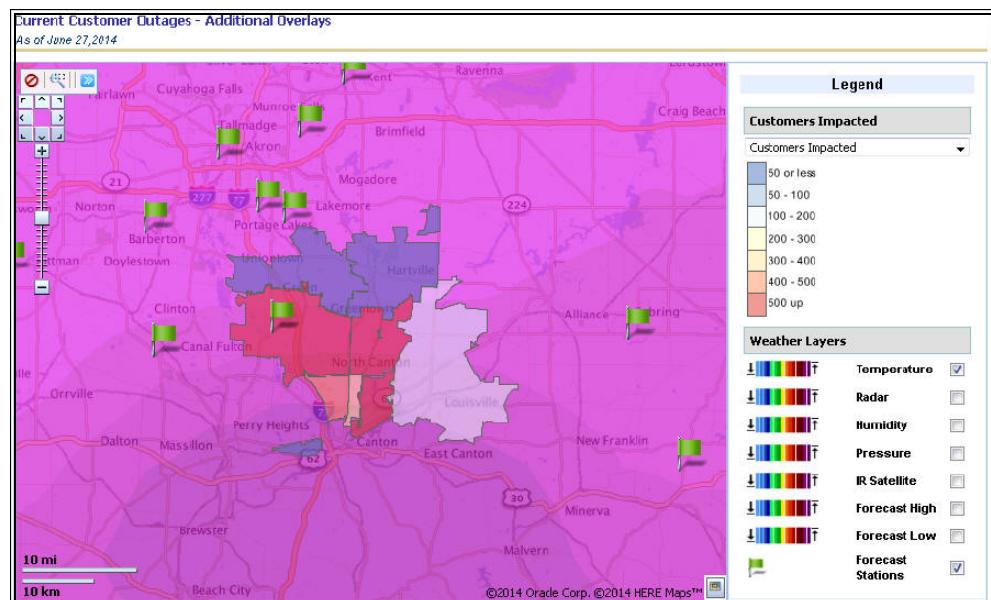
Note: The out-of-box weather integration code of Oracle Utilities Analytics works fine for Pulse APIs. If you want to opt for other weather information vendors, they would need to customize the code to work with other APIs. Every weather vendor will have their own set of APIs and the response format will vary from one to another. Hence, the weather integration code is different for every vendor.

Note: For additional details, refer to the white paper **Weather Integration using Third Party Weather APIs (Doc ID 1947857.1)** available at the My Oracle Support website (https://support.oracle.com).

Chapter 6

Managing Oracle Utilities Analytics

This chapter describes some of the important configuration settings that need to be applied to ensure that the Oracle Utilities Analytics product works smoothly. This chapter also provides recommendations on utilizing Oracle database partitioning to improve the performance of overall Oracle Utilities Analytics.

The chapter include:

*   Managing ELT Processes
*   Performance Recommendations

Managing ELT Processes

Oracle Data Integrator (ODI) with Oracle GoldenGate extract, load and transform data from the source database to the target database. Oracle GoldenGate captures the changes in the selected tables of the source schema and transfers the data to the replication schema. The tables in the replication schema are similar in structure to the source tables with a few additional columns added for history tracking and audit. Data retention in the replication schema is controlled by configuration stored in the metadata.

Based on the scheduler configuration, data from the replication schema is transferred by Oracle Data Integrator to the staging schema. The tables in the staging schema are similar to the tables in the target schema with a few additional columns for data transformations. There are no constraints on the staging tables. The foreign key mapping and other transformations are performed in the staging area. The data retention in the staging schema is controlled by configuration stored in the metadata.

From the staging schema, Oracle Data Integrator transfers data to the facts and dimensions in the target schema. Oracle Data Integrator also loads and refreshes the materialized views in the target schema. This chapter describes the following:

- Recommended Configuration
- Scheduling
- Monitoring the Jobs
- Debugging
- Handling the Errors
- Reloading the Data

Recommended Configuration

This sub-section covers the following:

- Database
- Estimating Connection Pool Size

Database

Setting	Recommended	Reason
Open Cursors	1000	Oracle Data Integrator based ELT processes utilize database connections to perform transformations to load data into the data warehouse. Depending on the number of processes running in parallel, additional connections are created to the database.
		Oracle Business Intelligence Enterprise Edition also uses the database connections to query and fetch data for the answers on various dashboards. Depending on the number of concurrent connected users, Oracle Business Intelligence Enterprise Edition spawns additional database connections.
		Considering these standard database accesses and other ad-hoc database access, the open cursors parameter should be set to a minimum of 1000. The higher values may be set based on the implementation usage.

WebLogic

Setting	Recommended	Reason
Connection Pool Size	200 (Default is 12)	Each Oracle Data Integrator job utilizes a connection from the pool and the connection is released when not in use by the job. Oracle Utilities Analytics ELT executes multiple jobs in parallel. The default configuration enables 6 parallel instances of each entity. This is particularly helpful during initial load when a large volume of historical data needs to be loaded into the data warehouse. Depending on the source system being configured, this can result in spawning up to 300 jobs. In case more than one source system is attached to Oracle Utilities Analytics, this number can increase further. Ensure that the connection pool size is increased appropriately when increasing the max_parallel configuration for an entity or when attaching more than one source system. In case the connection pool is not sized appropriately, you will notice that many jobs go into the waiting state until a connection becomes available.

Estimating Connection Pool Size

Use the query below to estimate the connection pool size. This should be executed after any changes in the max_parallel setting for the entities or after attaching a new source instance.

```
select round(
         greatest(
           sum(case when te.entity_type_flg in ('SCD1','SCD2')
                     then max_parallel
                     else 0
                 end)
          ,sum(case when te.entity_type_flg in
('FACT','RACC','SNAP')
                     then max_parallel
                     else 0
                 end)) *.6,-2)
estimated_connection_pool_size
   from mdadm.b1_target_entity te
      , mdadm.b1_job_config jc
  where jc.target_entity_id = te.target_entity_id;
```

Scheduling

In any data warehouse, the basic challenge is to get the data loaded quickly and efficiently whether it is the initial load or the incremental load. Data volumes are high during initial load and during incremental loads the volumes are considerably smaller. The Oracle Data Integrator based ELT processes utilize a time based slicing mechanism to split the load volumes into more manageable slices to process the initial as well as incremental loads. The metadata configurations allow you to control the size of the slice, parallelism of the loads, and much more. To ensure that the data loaded is always consistent, the process executions are governed by a set of rules. The following criteria are considered for the job execution:

- There should be no errors that need reprocessing.

- The maximum retries limit for the day should not be exceeded or reached.

- Tasks wait for the configured retry interval before submitting a retry for the job.

- Number of parallel jobs is always being limited to the maximum parallel configured.

- Jobs are not executed beyond the time of the most recent Oracle GoldenGate sync or to a specified cut off time, whichever is less.

- If a job is dependent on tables that are being synced by separate Oracle GoldenGate processes, then the common sync time of both processes is considered.

- If a scenario does not exist, then the jobs are not executed.

- An interface should be active.

- All dependencies should be run.

- The number of running/error instances of the job should be less than the maximum parallel executions allowed.

- If the instance is configured as run once, then it should not execute once it is successfully executed.

- If a job fails, it should be retried again until the maximum retries per day is reached. The interval between successive retries should be based on configuration.

- Oracle GoldenGate models comprising of the source tables used in the entity should have been synced. In case the sync timestamps vary across multiple models, then the maximum sync timestamp is used.

- The snapshot entities are executed on or after the snapshot end period.

- The schedule time can be used to stagger loads and distribute processing. A job is not being executed until the current time crosses the scheduled time.

    Note: Refer to the Configuring Target Entities section in the chapter 4: Oracle Utilities Analytics Administration Tool for where to schedule jobs.

All job executions are internally managed by the scenario named **B1_RUN_ALL**. This is scheduled using Oracle Data Integrator to run every 1-10 minutes per your requirement.

Monitoring the Jobs

As jobs keep running on a regular basis, they are monitored or tracked to ensure that the required performance parameters are met. Jobs are created so that they are capable of automatic re-execution and retries. However, the reasons for the failures should be examined and appropriate actions taken to resolve the issues. The following views are provided for achieving this:

View	Purpose
b1_jobs_vw	Provides a list of jobs executed with details of the slice start and end, scheduled execution time, actual start and end times, status, and record counts.
b1_config_vw	The consolidated view of the current configuration for all entities in the data warehouse.
b1_wait_reasons_vw	Erroneous configuration or other reasons may result in entities not running. Examine this view's results to see the reasons a job is prevented from executing.

Debugging

Oracle Data Integrator jobs are designed to make it easy to examine the data that was processed to determine issues in the processing pipeline. To do this, staging tables are utilized that retain data for a configured duration. Each execution results in some data that has been processed from the replication layer into the staging area and then on to the target entity. Each such set is stored in the staging table with the session number to identify the executing session.

If a job fails or if you wish to examine how data was before it was loaded into the target, you can query the associated staging tables to view the data.

The staging tables' names are derived by prefixing the target entity name with **"STG_"**.

Handling the Errors

There are three basic components utilized in the data processing pipeline:

- Oracle GoldenGate for replicating changes from the source to the target

- Oracle Data Integrator to load data from the replication layer to the target

- Oracle database to store and manage data in the different layers. Oracle data pump is utilized to perform the initial sync between the source database and the target database.

There is a possibility of errors occurring in one or more stages of the process. This section covers some of the details to help you identify the cause and provides pointers to take appropriate action to resolve the issues:

1. **Failure of Initial Sync Processes**: In case of an error, you can execute the initial sync process again and it resumes from where it left off. If the Oracle GoldenGate replication process is running while the initial sync is performed, it is possible that the process will fail due to the duplicate keys. Here are the steps to resolve the issue:

i) Stop the relevant replicat process.

ii) Delete the data from the tables, which are listed in the error.

iii) Execute the initial sync process again.

iv) After the initial sync has been successfully loaded, start the replicat processes.

2. **Failure of Oracle GoldenGate Processes**: Oracle GoldenGate processes can fail if the source or the target database goes down or network connectivity fails between the source and the target. To identify whether Oracle GoldenGate processes are running properly, you should login to the server and run the command **info all** on the Oracle GoldenGate command prompt. If any jobs are listed as ABENDED or STOPPED, verify the reason for the error by looking at the Oracle GoldenGate logs and start the jobs. Oracle GoldenGate processes will restart from the point of failure and continue the replication processes.

3. **Failure of Oracle Data Integrator Processes**: The Oracle Data Integrator processes have been designed to automatically retry for a configurable number of attempts per day. If the issue is related to a table spaces not being available or database connectivity issues, the jobs would automatically execute once the issue has been resolved.

If the issue resolution is anticipated to take few hours, it is advisable to disable the failing jobs for the duration of the fix.

Query the view B1_JOBS_VW in the metadata schema to identify if any jobs have failed. In case of a failure, the status_flg column will show 'E'. You can log into Oracle Data Integrator and filter by the session number (available in the jobs view) and look at the error encountered.

Any failures will rollback any changes to the target. A late arriving dimension will be tagged to a default key (-99) in the dimension and these will be reprocessed in subsequent loads. If the dimensional data has arrived into the warehouse, then the foreign key references will be corrected automatically.

Reloading the Data

Data warehouses are usually designed based on the assumption that data is added only once to the warehouse. Although rare, sometimes the data is reloaded again to reset specific entities. Oracle Utilities Analytics provides the functionality to reset and reload individual entities or all data associated with a specific instance of a source system:

• Reloading Replication

• Resetting an Entity

• Resetting an Instance

• Reloading the Extract Parameters

• Reloading the Bucket Configuration

Reloading Replication

Perform the following steps to reload the Replication. This should **not** be done in production environment as the historical data would be lost.

1. Login to **Oracle Data Integrator** client.

2. Navigate to the **Load Plans & Scenarios** section. Select the scenario named **B1_CLEANUP_REPLICATION**. Right-click and execute.

3. In the dialog box, select the **Context**.

For example, to reset the replication schema for Oracle Utilities Work and Asset Management product instance 1, select the context WAM1.

After the B1_CLEANUP_REPLICATION is executed successfully, the initial load from Source to Replication can be done by executing the B1_SYNC_CONTEXT.

Resetting an Entity

Perform the following steps to reset an Entity:

1. Login to **Oracle Data Integrator** client.

2. Navigate to the **Load Plans & Scenarios** section. Select the scenario named **B1_RESET_ENTITY**. Right-click and execute.

3. In the dialog box, select the **Context**.
   For example, to reset the fact for Oracle Utilities Device Management product instance 1, select **ODM1**.
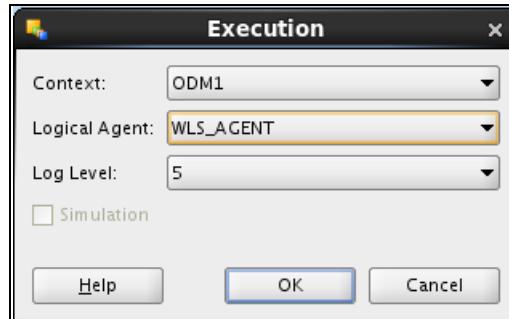
4. Uncheck **Last Value** checkbox, specify the variable value, and click on the **Name** field before clicking **OK**.
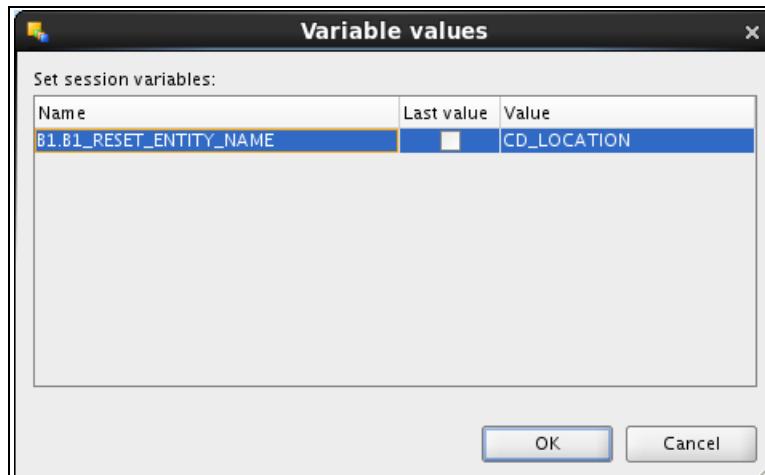
This cleans up the metadata entries for the executions of the ELT job for this specific entity. The data loaded in the particular table for the entity will also be clear. Note that the ELT job for this entity will now be in disabled state.

5. To start loading data again, enable the ELT job for the entity using the Oracle Utilities Analytics Administration Tool.

Resetting an Instance

Perform the following steps to reset an Instance:

> **Note**: The steps shown below to reset an instance should be used with care. The replication layer data should not be purged already, or else, the target does not load with the expected amount of data.

1. Login to the **Oracle Data Integrator** client.

2. Navigate to **Load Plans and Scenarios** section. Select the scenario named **B1_RESET_INSTANCE**. Right-click and execute.

3. In the dialog box, select the **Context**.
   For example, to reset all entities for Oracle Utilities Operational Device Management (ODM) product instance 1, select **ODM1**.

4. After verifying the configurations, enable all the entities to be load.

   > **Note**: When a reset instance is done on the data warehouse, the materialized view refresh jobs that are subsequently triggered can result in an error. It may result in a materialized view refresh path error. This is a known behavior of the Oracle database. To work around this issue, do a complete refresh of all the materialized views manually.

Reloading the Extract Parameters

If you need to modify the extract parameters defined on the source, perform the following steps:

1. Modify the parameters on the source application. Refer to the section Configuring Source for additional details.

2. Regenerate the replication views for the source instance for which parameters have been updated.

   For Unix:

   Perform the following steps:

   a. Navigate to the <Install_Dir>/bin directory.

   b. Add Java 8 to the path variable as shown below:

   ```
   export PATH=<Path_To_JDK_Directory>/bin:$PATH
   ```

   c. Run the **runviewGenerator.sh -c <Source Instance>**.

   For Windows:

   Perform the following steps:

   a. Navigate to the <Install_Dir>/bin directory.

   b. Add Java 8 to the path variable as shown below:

   ```
   set PATH= <Path_To_JDK_Directory>\bin;%PATH%
   ```

   c. Run the **runviewGenerator.cmd -c <Source instance>**.

   Where Source instance = Instance name given to the source in the **Configure Source** step mentioned above. For example: ODM1

3. Reload all the facts for the source products using the steps mentioned in the section Resetting an Entity.

Reloading the Bucket Configuration

Several key performance indicators in Oracle Utilities Analytics look at the measurement values and classify the value into an age range. The analysts can use these metrics to review the ages classified into different groups, such as 0-30 days, 30-90 days, or 90+ days. Such age ranges configuration is usually done before the implementation, but there may be scenarios where you would like to change these age ranges.

All of the ELT jobs that load these bucket configurations into the warehouse are set to be initial load only. Any incremental changes to these buckets after the initial run is not captured in Oracle Utilities Analytics because any changes made to the existing bucket configuration can cause inconsistency to the loaded fact data. However, if there is a need to reconfigure the buckets, certain additional steps need to be carried out to pull in the new configuration into the warehouse and ensure that this is reflected in the star schema tables.

The exact steps vary based on how the bucket configurations are used in the star schema ELT jobs. The below sections covers the two possible scenarios:

- Bucket Dimensions

- Facts Referencing Bucket Configuration

The exact detail of which bucket configuration falls under scenario is discussed in the section Configuring Source.

Bucket Dimensions

The typical scenario is that the configured buckets are loaded into dedicated bucket dimensions. The ELT jobs for the facts reference these buckets dimensions and map a fact record to an appropriate bucket configuration record in the dimension. This is the typical scenario that most bucket configurations fall under.

To reload this type of bucket configurations, perform the following steps:

1. Make the necessary changes to the bucket configurations in the source application.

2. Identify the bucket dimensions for the updated bucket configurations. Refer to the section Configuring Source for details.

3. The data that has been loaded into the bucket dimensions needs to be cleared and reloaded with the updated configuration. Since the dimensions need to be reloaded, the related facts also need to be reloaded. This list can be identified using the star schema mapping details provided in the *Oracle Utilities Data Mapping Guides* for the respective source product. Follow the steps mentioned in the section Resetting an Entity to reload the bucket dimensions and the related facts.

4. Reset the fact tables and then reset the bucket dimensions.

5. When re-enabling the ELT jobs, enable the bucket dimensions and then the facts.

Facts Referencing Bucket Configuration

The second scenario is that bucket configurations can be referenced by the fact ELT directly. This is a special use of the bucket configurations where a fact ELT would try to calculate a measure value based on the configured bucket ranges. Few bucket configurations fall under this scenario.

To reload such type of bucket configurations, perform the following steps:

1. Make the necessary changes to the bucket configurations in the source application.
For this type of bucket configuration where there is no dedicated bucket dimensions, the data is loaded in the metadata table B1_RANGE_LOOKUP in the MDADM schema.

2. Delete the existing rows in the MDADM.B1_RANGE_LOOKUP table for the specific bucket whose configurations are being updated.
    The table below identifies the bucket business object name. This name is stored as the range code in the metadata table and can be used to identify the rows to be deleted.

Source Product	Bucket	Business Object Name	Fact Name
CC&B	PA Future Payment Age Configuration	C1-PAFuturePaymentAge	CF_PA, CF_PA_SNAP
CC&B	PP Future Payment Age Configuration	C1-PPFuturePaymentAge	CF_PAY_PLAN, CF_PAY_PLAN_SNAP
CC&B	SA Arrears Configuration	C1-SAArrearsBuckets	CF_ARREARS
WAM	MTBF/MTTR Buckets	W1-MTBFMTTRBuckets	CF_ASSET_SNAP, CF_LOCATION_ SNAP

3. Follow the steps mentioned below to reload the updated bucket configuration once the identified rows have been deleted from the metadata table:

    For Unix:

    a. Navigate to the <*Install_Dir*>/bin directory.

    b. Add Java 8 to the path variable as shown below:

        export PATH=<*Path_To_JDK_Directory*>/bin:$PATH

    c. Run the runviewGenerator.sh -c <Source Instance>.

    For Windows:

    a. Navigate to the <*Install_Dir*>/bin directory.

    b. Add Java 8 to the path variable as shown below:

        set PATH= <*Path_To_JDK_Directory*>\bin;%PATH%
    c. Run the runviewGenerator.cmd -c <*Source instance*>.

4. Identify the affected facts for which the bucket configurations have been updated using the above table. Refer to the section Configuring Source for details.

5. The data that had been loaded into these facts would have been using the older bucket configuration, which gets invalidated when the configuration changes; therefore, the fact data needs to be reloaded. Follow the steps mentioned in the section Resetting an Entity to reload the related facts.

Dimensions Referencing Bucket Configurations

The third scenario is that bucket configurations can be referenced by the dimension ELT directly. These would be like any other dimension with attribute columns, but will also have the bucket range values. This is a special use of the bucket configurations where the dimension ELT would try to map a value in the dimension (for example, condition score of an asset) to a matching bucket range it falls into and populate the bucket range details in the dimension. Few bucket configurations fall under this scenario.

To reload such type of bucket configurations, perform the following steps:

1. Make the necessary changes to the bucket configurations in the source application.
    For such type of bucket configurations the data is loaded in the metadata table B1_RANGE_LOOKUP in the MDADM schema.

2. Delete the existing rows in the MDADM.B1_RANGE_LOOKUP table for the specific bucket whose configurations are being updated.
   The following table identifies the bucket business object name. This name is stored as the range code in the metadata table and can be used to identify the rows to be deleted.

Source Product	Bucket	Business Object Name	Fact Name
WAM	Asset Condition Score Buckets	W1-AssetConditionScoreBuckets	CD_UTIL_ASSET
WAM	Asset Criticality Buckets	W1-AssetCriticalityBuckets	CD_LOCATION
WAM	Work Priority Buckets	W1-WorkPriorityBuckets	CD_WO

3. Follow the steps below to reload the updated bucket configuration once the identified rows have been deleted from the metadata table:

   For Unix:

   a. Navigate to the *<Install_Dir>*/bin directory.

   b. Add Java 8 to the path variable as shown below:

   ```
   export PATH=<Path_To_JDK_Directory>/bin:$PATH
   ```
   c. Run the runviewGenerator.sh -c <Source Instance>.

   For Windows:

   a. Navigate to the *<Install_Dir>*/bin directory.

   b. Add Java 8 to the path variable as shown below:

   ```
   set PATH= <Path_To_JDK_Directory>\bin;%PATH%
   ```
   c. Run the runviewGenerator.cmd -c <Source instance>.

4. Identify the affected dimensions for which the bucket configurations have been updated using the above table. Refer to the section Configuring Source for details.

5. The data that had been loaded into these dimensions would have been using the older bucket configuration, which gets invalidated when the configuration changes; therefore, the dimension data needs to be reloaded. Since the dimensions are referenced by facts, the dependent facts on the affected dimensions have to be reloaded. Follow the steps mentioned in the section Resetting an Entity to reload the related dimensions and facts.

Performance Recommendations

This section discusses how to configure the database for better system performance, including:

- Partitioning
  - Partitioning of the Snapshot Facts
  - Partitioning of the Accumulation Facts
- Parallelism
- Optimizing the Top N Answers
- Archiving the Historical Data

Partitioning

Partitioning helps to scale a data warehouse by dividing the database objects into smaller pieces, enabling access to smaller, more manageable objects. Having direct access to the smaller objects addresses the scalability requirements of the data warehouses.

It takes longer to scan a big table than it takes to scan a small table. Queries against partitioned tables may access one or more partitions that are small in contrast to the total size of the table. Similarly, queries may take advantage of partition elimination on indexes. It takes less time to read a smaller portion of an index from the disk than to read the entire index. The index structures that share the partitioning strategy with the table, such as local partitioned indexes, can be accessed and maintained on a partition-by-partition basis.

The partitioning of the facts is explained below in:

• Partitioning of the Snapshot Facts

• Partitioning of the Accumulation Facts

Partitioning of the Snapshot Facts

Most of the out-of-the-box snapshot facts are delivered as Monthly snapshots. This means that the fact will have rows for each month from the beginning of the implementation.

All snapshot facts should be partitioned by the snapshot date. Listed below are the Snapshot facts and the column on which the partition should be created. When converting an existing fact into a partitioned fact ensure that the indexes, which do not have the partition key columns, are created as the local indexes. Ensure that the partitions are also created for the future periods so that the future snapshot loads do not fail while adding data.

For monthly snapshots, primary partition based on the year and sub-partition based on the calendar month work well. For snapshots that are not monthly, different set can be used.

Source Product	Fact	Column	Partition Type
CCB	CF_ARREARS	SNAPSHOT_DT	List
CCB	CF_PAY_PLAN_SNAP	SNAPSHOT_DT	List
CCB	CF_PA_SNAP	SNAPSHOT_DT	List
NMS	CF_CITY_OUTG	SNAPSHOT_DT	List
NMS	CF_CTRL_ZONE_OUTG	SNAPSHOT_DT	List
NMS	CF_FEEDER_DLVRD_LOAD	SRC_DTTM	List
NMS	CF_OUTG	SNAPSHOT_DTTM	List
WAM	CF_OPR_DEVICE_SNAP	SNAPSHOT_DT	List
WAM	CF_ASSET_SNAP	SNAPSHOT_DT	List
WAM	CF_LOCATION_SNAP	SNAPSHOT_DT	List

Partitioning of the Accumulation Facts

The accumulation facts can also be partitioned based on a date key. However, the data in the accumulation facts will be spread across multiple dates and a list based partition will not work for the accumulation facts. A range based partitioning scheme should be employed on the primary date_key column in the specific fact.

The following query gets the start and end range values from the CD_DATE dimension. Using this range, values for individual partitions can be defined.

```
select abs_month_nbr  part_name_suffix
     , min(date_key) part_range_start
     , max(date_key) part_range_end
  from dwadm.cd_date
  group by abs_month_nbr
  order by 1;
```

Parallelism

The database can take advantage of the distinct data sets in separate partitions if you use parallel execution to speed up queries, Data Manipulation Language (DML), and Data Definition Language (DDL) statements. Individual parallel execution servers can work on their own data sets, identified by the partition boundaries.

The parallel execution enables the application of multiple CPU and I/O resources to the execution of a single database operation. It dramatically reduces response time for data-intensive operations on large databases typically associated with a Decision Support System (DSS) and data warehouses. You can also implement parallel execution on an Online Transaction Processing (OLTP) system for batch processing or schema maintenance operations, such as index creation. Parallel execution is also called 'Parallelism'. Parallelism involves breaking down a task so that instead of one process doing all of the work in a query, many processes do part of the work at the same time. For example, when four processes combine to calculate the total sales for a year, each process handles one quarter of the year instead of a single process handling all four quarters. The improvement in performance can be quite significant. The parallel execution improves processing for:

• Queries requiring large table scans, joins, or partitioned index scans

• Creation of large indexes

• Creation of large tables (including materialized views)

• Bulk insertions, updates, merges, and deletions

> **Note**: For details on parallelism, partitioning, and other performance enhancement options, refer to *Oracle Database VLDB and Partitioning Guide 11g Release 2*.

Optimizing the Top N Answers

The Top N charts have to go through millions of records to find out the Top N objects that meet the criteria. Top N materialized views rearrange and partition the data so that the data reads are optimal. However, at times, depending on the amount of data, additional configuration may be required to reduce the size of the data set that is being scanned in order to identify the Top N objects.

For example, a large sized utility with 6 million customers may have around 6 million service points in their Oracle Utilities Meter Data Management application. This means that the Consumption fact in the Oracle Utilities Meter Data Management star schema may have several million records for every snapshot month. So, for each month level partition on the **Consumption Detail Level** materialized view, there will be several million records. The detail level pages (Top N Analysis and Unreported Usage Details) on the **Usage Summary** dashboard under Oracle Utilities Meter Data Analytics tries to access a huge volume of data, which can result in sluggish performance sometimes. The recommendation in such scenarios is to make a key prompt filter such as, **'City'** as mandatory. This ensures the report looks at a smaller data set; thereby, improving the report performance. You can choose to make any set of filters mandatory as per your data requirement.

Archiving the Historical Data

Assuming that the facts are partitioned as per the recommendations, it is now possible to archive historical data by selecting specific partitions and archiving them to a backup. Archival rules should be based on your data retention policies. This can be set up as a scheduled activity.