

# **Oracle Utilities Load Profiling and Settlement**

Installation and Configuration Guide

Release 1.6.1.23 for Windows

**E18209-24**

December 2018

(Revised April 2020)

Oracle Utilities Load Profiling and Settlement/Load Profiling and Settlement Installation and Configuration Guide, Release 1.6.1.23 for Windows

E18209-24

Copyright © 1999, 2018 Oracle and/or its affiliates. All rights reserved.

Primary Author: Lou Proseri

Contributor: Steve Pratt

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

## **NOTIFICATION OF THIRD-PARTY LICENSES**

Oracle Utilities software contains third party, open source components as identified below. Third-party license terms and other third-party required notices are provided below.

**License:** Apache 1.1

**Module:** xercesImpl.jar, xalan.jar

Copyright © 1999-2000 The Apache Software Foundation. All rights reserved.

Use of xercesImpl and xalan within the product is governed by the following (Apache 1.1):

(1) Redistributions of source code must retain the above copyright notice, this list of conditions and the disclaimer below. (2) Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the disclaimer below in the documentation and/or other materials provided with the distribution. (3) The end-user documentation included with the redistribution, if any, must include the following acknowledgment: “This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).” Alternately, this acknowledgment may appear in the software itself, if and

wherever such third-party acknowledgments normally appear. (4) Neither the component name nor Apache Software Foundation may be used to endorse or promote products derived from the software without specific prior written permission. (5) Products derived from the software may not be called “Apache”, nor may “Apache” appear in their name, without prior written permission.

THIS SOFTWARE IS PROVIDED “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**License:** Paul Johnston

**Modules:** md5.js

Copyright (C) Paul Johnston 1999 - 2002

Use of these modules within the product is governed by the following:

(1) Redistributions of source code must retain the above copyright notice, this list of conditions and the disclaimer below. (2) Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the disclaimer below in the documentation and/or other materials provided with the distribution. (3) Neither the component name nor the names of the copyright holders and contributors may be used to endorse or promote products derived from the software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**License:** Tom Wu

**Module:** jsbn library

Copyright © 2003-2005 Tom Wu. All rights reserved

Use of this module within the product is governed by the following:

(1) Redistributions of source code must retain the above copyright notice, this list of conditions and the disclaimer below. (2) Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the disclaimer below in the documentation and/or other materials provided with the distribution.

THE SOFTWARE IS PROVIDED “AS-IS” AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL TOM WU BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.



---

---

# Contents

## Contents

### Chapter 1

Overview.....	1-1
Installation and Configuration Overview .....	1-2
What is this book? .....	1-3

### Chapter 2

<b>A Day in the Life of Oracle Utilities Load Profiling and Settlement .....</b>	<b>2-1</b>
Oracle Utilities Load Profiling and Settlement Processes.....	2-2
Profiling .....	2-2
Aggregation.....	2-4
Load Settlement .....	2-7
Financial Settlement.....	2-8
Monthly Settlement Example .....	2-9
Step One: Import Data .....	2-9
Step Two: Validate Data .....	2-9
Step Three: Calculate Class Profiles for Use in Settlement .....	2-9
Step Four: Pre-Aggregate Meter Reads .....	2-9
Step Five: Profile Available Aggregated Meter Read Records .....	2-10
Step Six: Profile Remaining Meter Reads, Estimate for Missing Reads .....	2-10
Step Seven: Interval Metered Customers .....	2-10
Step Eight: Aggregation .....	2-10
Step Nine: UFE.....	2-10
Step Ten: Imbalances and Financial Settlement.....	2-11
Step Eleven: Service Coordinator Billing .....	2-11
Final Settlement.....	2-11

### Chapter 3

<b>Oracle Utilities Load Profiling and Settlement Network Environment and Database Installation.....</b>	<b>3-1</b>
The Oracle Utilities Load Profiling and Settlement Installation Package.....	3-2
Oracle Utilities Load Profiling and Settlement Network Environment .....	3-3
Client Workstation - C/S.....	3-4
Client Workstation - Web.....	3-5
Database Server.....	3-6
Application/Batch Processing Server.....	3-7
Web Server(s).....	3-8
Oracle Utilities Load Profiling and Settlement Database Installation.....	3-9
Installation Requirements .....	3-9
Installing the Database .....	3-9
Verifying the Database.....	3-10

### Chapter 4

<b>Installing the Oracle Utilities Load Profiling and Settlement Application Software.....</b>	<b>4-1</b>
Installing the Oracle Utilities Load Profiling and Settlement Software .....	4-2
Installing in Conjunction with the Energy Information Platform .....	4-2

Installing After Installation of the Energy Information Platform.....	4-2
Setting Up Configuration Files on the Web Server.....	4-3
Setting Up Configuration Files on Application Servers and Workstations.....	4-5
Installing Oracle Utilities Load Profiling and Settlement Database Records and Rate Forms .....	4-6
<b>Chapter 5</b>	
<b>Setting Up Oracle Utilities Load Profiling and Settlement Database Records .....</b>	<b>5-1</b>
Setting Up Oracle Utilities Load Profiling and Settlement Lookup Tables .....	5-2
Aggregation Type Table.....	5-2
Distribution Company Table .....	5-2
Loss Class Table.....	5-2
Market Table.....	5-3
Method Table .....	5-3
Profile Class Table .....	5-3
REP Table.....	5-3
Scheduling Coordinator Table .....	5-3
Service Table.....	5-4
Settlement Type Table.....	5-4
Superzone Table.....	5-4
Zone Table.....	5-4
<b>Chapter 6</b>	
<b>Configuring Oracle Utilities Load Profiling and Settlement .....</b>	<b>6-1</b>
Customizing Oracle Utilities Load Profiling and Settlement Processes .....	6-2
Customizing Parameters for Settlement Processing .....	6-3
Customize Parameters in Rate Schedule XSD .....	6-3
Customize Parameters in Oracle Utilities Load Profiling and Settlement Rules Language .....	6-6
<b>Chapter 7</b>	
<b>Using the Proxy Day Extension with Oracle Utilities Load Profiling and Settlement.....</b>	<b>7-1</b>
How the Proxy Day Extension DLL Works.....	7-2
Some Important Notes .....	7-2
Required Components.....	7-4
Proxy Day Extension DLL Inputs .....	7-5
Triggering the Proxy Day Extension DLL with the Rules Language .....	7-10
Data Returned from Proxy Day .....	7-12
Proxy Day Return Codes.....	7-13
Proxy Day Tests and DST .....	7-17
Magnitude Test.....	7-17
Shape Test.....	7-17
<b>Chapter 8</b>	
<b>Configuring Oracle Utilities Load Profiling and Settlement Security .....</b>	<b>8-1</b>
Oracle Utilities Load Profiling and Settlement Security.....	8-2
Profiling and Settlement Features .....	8-2
Important Notes about Assigning Oracle Utilities Load Profiling and Settlement Permissions .....	8-2
<b>Appendix A</b>	
<b>Oracle Utilities Data Repository Load Profiling and Settlement Database Schema.....</b>	<b>A-1</b>
Oracle Utilities Load Profiling and Settlement Database Schema, page 1 .....	A-2
Oracle Utilities Load Profiling and Settlement Database Schema, page 2 .....	A-3
<b>Index</b>	

# Chapter 1

---

## Overview

This chapter provides an overview of the installation and configuration of the Oracle Utilities Load Profiling and Settlement web application, including:

- **Installation and Configuration Overview**
- **What is this book?**

## Installation and Configuration Overview

Installing and configuring Oracle Utilities Load Profiling and Settlement involves the following steps:

- Setup the Oracle Utilities Load Profiling and Settlement network environment and the Oracle Utilities Data Repository as described in **Chapter 3: Oracle Utilities Load Profiling and Settlement Network Environment and Database Installation**.
- Install the Oracle Utilities Load Profiling and Settlement Database as described in **Chapter 3: Oracle Utilities Load Profiling and Settlement Network Environment and Database Installation** and **Chapter 3: Oracle Utilities Data Repository Schema Creation** of the *Oracle Utilities Energy Information Platform Installation Guide*.
- Install workstation and application server applications used by Oracle Utilities Load Profiling and Settlement on client machines as described in **Chapter 4: Installing the Oracle Utilities Load Profiling and Settlement Application Software**.
- Install Oracle Utilities Load Profiling and Settlement Web components on web server as described in **Chapter 4: Installing the Oracle Utilities Load Profiling and Settlement Application Software**.
- Install Oracle Utilities Load Profiling and Settlement database records and rate forms as described in **Chapter 4: Installing the Oracle Utilities Load Profiling and Settlement Application Software**.
- Set up additional database records as described in **Chapter 5: Setting Up Oracle Utilities Load Profiling and Settlement Database Records**.
- Configure Oracle Utilities Load Profiling and Settlement as necessary as described in **Chapter 6: Configuring Oracle Utilities Load Profiling and Settlement**.
- Set up and configure the Proxy Day Extension DLL for use with Oracle Utilities Load Profiling and Settlement as described in **Chapter 7: Using the Proxy Day Extension with Oracle Utilities Load Profiling and Settlement**.
- Set up and configure security Roles and Permissions for use with Oracle Utilities Load Profiling and Settlement as described in **Chapter 8: Configuring Oracle Utilities Load Profiling and Settlement Security**.



---

## What is this book?

The *Oracle Utilities Load Profiling and Settlement Installation and Configuration Guide* describes how to install and configure Oracle Utilities Load Profiling and Settlement, including the following:

- **Chapter 1: Overview** (this chapter) provides an overview of the Oracle Utilities Load Profiling and Settlement installation and configuration process
- **Chapter 2: A Day in the Life of Oracle Utilities Load Profiling and Settlement**
- **Chapter 3: Oracle Utilities Load Profiling and Settlement Network Environment and Database Installation** describes the network environment used by Oracle Utilities Load Profiling and Settlement, and how to install the Oracle Utilities Load Profiling and Settlement database.
- **Chapter 4: Installing the Oracle Utilities Load Profiling and Settlement Application Software** describes how to install Oracle Utilities Load Profiling and Settlement application software on an application server, on a web server, and on client machines.
- **Chapter 5: Setting Up Oracle Utilities Load Profiling and Settlement Database Records** describes how to set up and configure database records used by the Oracle Utilities Load Profiling and Settlement application.
- **Chapter 6: Configuring Oracle Utilities Load Profiling and Settlement** describes how to add customized business logic to Oracle Utilities Load Profiling and Settlement.
- **Chapter 7: Using the Proxy Day Extension with Oracle Utilities Load Profiling and Settlement** describes how to use the Proxy Day Extension DLL with Oracle Utilities Load Profiling and Settlement.
- **Chapter 8: Configuring Oracle Utilities Load Profiling and Settlement Security** describes how to configure security for use with Oracle Utilities Load Profiling and Settlement.
- **Appendix A: Oracle Utilities Data Repository Load Profiling and Settlement Database Schema** provides a database schema diagram of the database tables used with Oracle Utilities Load Profiling and Settlement.



# Chapter 2

---

## A Day in the Life of Oracle Utilities Load Profiling and Settlement

This chapter describes the processes performed by Oracle Utilities Load Profiling and Settlement, and outlines the typical tasks performed by Oracle Utilities Load Profiling and Settlement on a typical day when implemented to perform monthly settlement. This includes:

- **Oracle Utilities Load Profiling and Settlement Processes**
  - Profiling
  - Aggregation
  - Load Settlement
  - Financial Settlement
- **Monthly Settlement Example**
  - Step One: Import Data
  - Step Two: Validate Data
  - Step Three: Calculate Class Profiles for Use in Settlement
  - Step Four: Pre-Aggregate Meter Reads
  - Step Five: Profile Available Aggregated Meter Read Records
  - Step Six: Profile Remaining Meter Reads, Estimate for Missing Reads
  - Step Seven: Interval Metered Customers
  - Step Eight: Aggregation
  - Step Nine: UFE
  - Step Ten: Imbalances and Financial Settlement
  - Step Eleven: Service Coordinator Billing
  - Final Settlement

# Oracle Utilities Load Profiling and Settlement Processes

This section describes the business processes performed by Oracle Utilities Load Profiling and Settlement, including:

- **Profiling**
- **Aggregation**
- **Load Settlement**
- **Financial Settlement**

## Profiling

Load profiling (or simply profiling) is the market technique that allows the majority of end use customers to participate in an interval market without requiring interval metering to be installed at each end use customer. Profiling allows entities without metered interval data to participate in an interval market by assigning to them estimated interval data. This estimated data is aggregated with actual interval metered data to arrive at an estimate of interval data for a market participant.

### Historical Class Profiles

Historical class profiles are central to the profiling process. Historical class profiles are load profiles consisting of historical interval data that is representative of a specific customer class or type. For instance, if residential customers are considered a ‘customer class,’ a Residential class profile might represent the average load shape of residential customers for a given time period (a day, a week, a month, etc.). Creating class profiles usually involves examining a statistical sample of the target population that had been outfitted with interval metering to create a curve that represents the average customer for the class. This is traditionally called load research.

Class profiles are typically stored in a historical database (this is also referred to as load research data). This database can then be used in the creation of load profiles to be used in the settlement process (see **Load Profiling Overview** on page 2-2).

### Creating New Historical Class Profiles

As additional historical interval data is gathered, that data can be added to existing historical data to increase number and size of samples from which class profiles can be drawn. This allows for more accurate sampling and potentially more accurate profiling.

### Load Profiling Overview

Oracle Utilities Load Profiling and Settlement can employ a variety of profiling methods and apply those profiles to individual accounts or account aggregations (see **Aggregation** on page 2-4) to arrive at profiled load estimations for customers, segments, suppliers, or schedule coordinators, or to reconcile the estimations to the system load. The profiling routines encoded in Oracle Utilities Load Profiling and Settlement settlement schedules take monthly scalar meter reads and develop customer-specific usage profiles by applying the energy to the appropriate profile template for the meter reading interval. These profiles can be aggregated immediately or stored to the database as individual customer interval data. In addition, Oracle Utilities Load Profiling and Settlement can perform usage factor profiling and any other user-defined calculations.

Load profiling can be divided into two efforts: profile creation and profile application. Each is described below.

### Profile Creation

Profile creation is the process of creating interval data that represents the usage of the average customer in a particular subset of the population on an hourly or sub-hourly basis. This usually involves selecting an appropriate class profile from historical data and adjusting that profile as necessary for use in the settlement process. Selecting the appropriate historical class profiles to be used in the settlement process can be based on any number of factors, including:

- Customer Class (Residential, Commercial, Industrial, etc.)
- Type of Service
- Day of Week
- Holidays
- Season (Winter, Summer, etc.)
- Weather Data (Temperature, wind speed, relative humidity, etc.).

The methods used to create the class profile that will be used for settlement differ by utility and even by rate class within the utility, and are largely influenced by existing load research and data collection efforts. Every profile creation method depends on load research, whether historical, borrowed, or current, to arrive at a profile that will be used in settlement.

Oracle Utilities Load Profiling and Settlement has the capability to perform a variety of profile creation methods, including the following.

**Static Load Profiling:** Static load profiling involves the use of historical class load research. The historical data is shifted forward in time as a unit, adjusting for the day of the week and holidays as necessary. This data may represent a historical year, or an average of several years. This data is readily available, but it is not adjusted to represent the day of interest in any way other than matching the calendar day and holidays when necessary.

**Adjusted Static Load Profiling:** Adjusted static load profiling involves adjusting the static load profiles for the day of interest to more closely represent the characteristics of the day of interest. For example, the class profiles can be adjusted by hour by a ratio of the current system load to the historic system load. This method assumes that the class load shapes are closely related to the system load, and adjusts the profiles from the static day, scaling them up or down to more closely approximate the current system load. Weather data can also be used to adjust static load profiles for each settlement day.

**Proxy Day Load Profiling:** Proxy day load profiling, like static profiling, relies on historical load research. However, it differs from the static methods described above in that the data for only one specific day is moved forward in time to represent another specific day. Readily available characteristics from the day of interest, such as weather, system load, day type, etc., that are assumed to reliably and predictably affect load, are used to mine a historical database of class load shapes and select the same related information for the historical days. The day of data with the closest match to the current characteristics is selected as the proxy day, and is shifted forward in time to the day of interest.

**Dynamic Modeling Load Profiling:** Dynamic modeling is the process of creating a model of the class load shapes from historical data and readily available related information, such as weather, day type, etc. This method is similar to the proxy day method in that it creates load shapes for the day of interest based on readily available correlated data from the day of interest. It differs from proxy day in that instead of borrowing a day of data from history that most closely matches the characteristics from the day of interest, it creates each hour in the day based on modeling the historical data and correlated data.

**Dynamic Load Profiling:** Dynamic load profiling is the process of developing class load profiles for the day of interest by querying all of the sample meters, importing, validating, editing, and estimating all sample data, performing statistical analysis on the sample data to arrive at class load profiles, and using the profiles in the settlement process. Dynamic load profiling can be considered traditional load research on an extremely accelerated timeframe. Dynamic load profiling can be performed using the Statistical Analysis Extension of Oracle Utilities Load Analysis, (available under separate license).

## Profile Application

Profile application is the process of creating interval data for an individual entity or group of entities by scaling the appropriate class load profile to match the usage or estimated usage of the entity for a given period of time. In short, it is the process of using the class load profile to create data that more accurately represents the usage of any member or group of members of that class. Profile application methods can differ by implementation and settlement process. For example, profile application for a settlement day that is one day from the current day will not have the benefit of any meter readings on the class members. Their usage must be estimated based on historical data and conditions of the settlement day. However, if the settlement day is 45 or 60 days in the past, all meter reads with usage for the settlement day are available, and a different process can be used that utilizes the available metered data for that day.

Oracle Utilities Load Profiling and Settlement has the power and flexibility to allow a variety of profile application methods. Through the use of the Oracle Utilities Rules Language, users can continue to modify, change, and enhance their profile application methods to best suit their needs.

**Unadjusted Profile Application:** Unadjusted profile application involves the assumption that each member of a profile class uses the same amount of energy. A supplier estimate is generated by counting the number of customers in each profile class, and multiplying the class average load profile by that customer count. This method has the potential for significant bias, in that if a supplier's customers deviate from the average, their estimate can be significantly off.

**Usage Factor Profile Application:** A usage factor is a scalar representation of the relative size of a customer compared to the average customer in its class. For example, a usage factor could be the average usage per day for the customer divided by the average usage per day for the class for the same period in time. An average customer would have a usage factor of one. An account-specific profile can be generated by multiplying the class average load profile by the account's usage factor.

**Scalar Energy Reading Profile Application:** Account-specific profiles can also be generated by using the account's meter reading and the class load profile. This method is typically used in the final settlement stage, when all meter readings covering the settlement period have been processed. An account-specific profile is generated by multiplying the class profile for the same time period as the account meter read by a ratio of the energy in the account meter read, divided by the energy in the class profile for the same period of time as the account meter read. The resulting load shape maintains the shape of the class load profile, but takes on the scale of the account meter read.

## Aggregation

Data Aggregation involves summing real or assigned (profiled or deemed) interval data to a user-defined aggregation level for further processing. Aggregation may occur at a variety of levels for a variety of reasons, including reporting requirements and applying losses and other factors at an aggregated level up to the market participation level. Aggregation is typically used to build ground-up estimates of load for energy market participants for forecasting, scheduling, and settlement.

Aggregation techniques can also be used to pre-aggregate scalar data before profiling based on common characteristics. This allows users to quickly perform settlement for large numbers of end use customers by grouping energy readings or usage factors by like groups, and processing the groups as you would each individual in the group.

## Losses and Unaccounted for Energy (UFE)

Besides summing data at a given level, data aggregation is also used when calculating losses and Unaccounted for Energy (UFE).

Markets typically operate using generation-level determinants, meaning that the individual metered entity data must be adjusted for system losses in order to compare the market participant load to the system load, market participant schedules, etc. Loss calculations can vary from simple to complex, ranging from an annual percentage by rate class or voltage level, to seasonal or time of

use loss factors, to hourly dynamic loss factor calculations based on system conditions. Loss factors can be applied at the metered level or any aggregation level, as needed. In addition, losses can be applied to generators in markets where generation losses are calculated and used.

Unaccounted for Energy (UFE) is calculated by comparing the known system hourly load to the sum of all market participant hourly estimates from the load profiling, aggregation, and loss adjustment processes. Because the market participant estimates will never perfectly match the system load, these estimates are fine-tuned by allocating a portion of the total UFE to each estimate. Allocation methods differ, but in general each market participant is allocated a portion of the system UFE for each hour based on the ratio of that particular estimate's load in that hour to the total of all estimated loads in that hour. This process forces the total of all estimates to match the system load. Some markets use different UFE allocation methods, including differential allocation of UFE to interval metered loads apart from profiled loads, and adjusting the profile methodologies to minimize UFE where possible.

## **Aggregation Groups**

Because aggregation involves summing up interval data to a specified level for processing, it's important to understand what is meant by the term "aggregation level" or "aggregation group." Put simply, an aggregation group is any point along the distribution chain to which entities below that point are summed. For instance, if you were going to sum up all the accounts that are related to a specific bus, that bus would be an aggregation group. Similarly, if you were going to aggregate all the energy used by all the accounts for a given Energy Service Provider (ESP), that ESP would be an aggregation group. In general, aggregation is usually performed at a level below you on the distribution chain from generators to end users.

The entities and/or market participants that serve as aggregation groups are specific to the particular area and market in which your utility is operating, but can include the following:

### **Energy Service Providers (ESP)**

Energy Service Providers (ESPs) sell power to customers. They may generate it and/or buy it from other sources. ESPs are also referred to as Retail Energy Providers (REPs) or simply suppliers.

### **Distribution Companies (DisCo)**

A distribution company is responsible for the delivery of power to all customers in its service territory (Jurisdiction), regardless of who the customers have selected as their suppliers. The DisCo owns and operates the distribution lines that bring electricity into homes and businesses, and is responsible for metering, billing, and service calls. However, a DisCo may or may not be affiliated with a customer's power supplier.

Distribution companies are also referred to as Operating Companies, Utility Distribution Companies (UDCs), or Local Distribution Companies (LDCs).

### **Schedule/Service Coordinators (SC)**

Schedule Coordinators (or Service Coordinators) are responsible for creating schedules of energy usage that represent the estimated usage for a future time period (most often one day) for a given segment of the market. Schedules are most often based on results generated during the load profiling process, but can also be based on historical usage information and other data.

### **Independent System Operators (ISO)**

An Independent System Operator (ISO) is a non-profit corporation that manages the regional power grid to coordinate the generation and transmission of energy. Its goal is to guarantee a safe and reliable power supply, and to ensure that all suppliers get nondiscriminatory access to the transmission system (while the ISO controls the power grid, commercial entities still own it). In the U.S. market, ISOs are regulated by the Federal Energy Regulatory Commission (FERC).

Data concerning the various aggregation groups used in the settlement process is stored in the Aggregation Group and Account Service History tables in the Oracle Utilities Data Repository. The specific information the database stores about each aggregation group depends on its role in the settlement process. For example, a Distribution Company might only aggregate data at the Bus, REP/ESP, or DisCo levels, and an ISO might perform aggregation at any level up to and including the ISO level.

### **Aggregation Types**

In addition to aggregating data by aggregation groups, it is useful to be able to aggregate by other factors, such as customer class, voltage class, or loss class. Some common aggregation types are described below. Note that these are provided as examples of common aggregation types only. Specific markets may require aggregation types other than those listed here.

#### **Customer Class**

A customer class represents a 'type' of customer or account. Three common customer classes include residential, commercial, and industrial, but others are also prevalent. In addition, there may be multiple types of residential, commercial, or industrial classes, representing distinct types of accounts within each of those broad categories. For instance, residential customers might be divided into apartments, condominiums, and houses.

#### **Voltage Class**

A voltage class represents a group of customers or accounts based on the type of voltage used by the end users in that group. Voltage classes differ by market and region, but some common voltage classes include Transmission, Primary, and Secondary. Aggregating by voltage class is often performed for reporting purposes, to track usage of a given voltage type within the system.



**Loss Class**

A loss class represents a group of customers or accounts based on the type of loss (transmission loss, distribution loss, etc.) associated with those accounts. Loss is often related to voltage levels, and so correlations between loss classes and voltage classes are common. Aggregation by loss class is used when calculating losses in the overall settlement process, and allows different loss calculation algorithms to be used with different loss classes, as appropriate.

**Rate Class**

A rate class represents a group of customers or accounts based on pricing. Rate classes can be based on a number of characteristics, including usage, territory, or even voltage type. Aggregation by rate class can be used for reporting purposes, as well as when performing financial settlement (see **Financial Settlement** on page 2-8).

**Load Settlement**

Load settlement is the process of deriving billing determinants to be used in supplier billing based on the results of the load profiling and data aggregation processes. The load settlement process also involves calculating imbalances.

**Calculating Imbalances**

An imbalance is the difference between two load shapes representing different estimates of the same period, typically a scheduled load shape and an actual load shape. Imbalances are typically used when money has changed hands based on an estimate that has been improved or finalized by the settlement process. Because the second estimate has more accurate data, the allocations performed by the first process have to be modified.

Imbalance shapes can be further divided by applying bands and TOU components. Bands are error tolerances around zero that allow for differential pricing of the imbalance curve by how well or poorly the scheduled estimate matched the actual usage. Time of Use periods are often defined, allowing a higher penalty for faulty forecasting during higher-priced, on-peak periods. Whether an imbalance calculation is performed and how it is performed varies greatly from market to market.

## Financial Settlement

The purpose of creating interval data for each market participant is that the costs of procuring energy in the market, as well as all associated costs such as ancillary services, fees, and penalties, must be billed to the supplier. Financial settlement can typically occur at the utility or at the Independent System Operator (ISO). Financial settlement involves calculations performed with market participant load, UFE, and imbalance from the load settlement process, market prices for energy, ancillary services markets, bilateral contracts, and any other information needed to determine costs. This process resembles complex billing in that although there is not a significant amount of data being processed, there are usually very complex rules to follow, and intricate calculations that may need to be performed.

Financial settlement is the process of calculating charges for energy and services based on settlement data and determinants derived from the profiling, aggregation, and load settlement processes. Financial settlement can involve a number of factors, including market participant load, Unaccounted for Energy (UFE), imbalances, market prices, and ancillary services. Each of these factors is discussed below.

### Market Participant Load

Market participant load represents the load for each participant in the market, and is the load shape used in the financial settlement/billing calculations. This generally includes the participant's load (scheduled or deemed), application of any loss factors, and an appropriate portion of the total system UFE.

### UFE

Unaccounted for Energy (UFE) is calculated by comparing the known system hourly load to the sum of all market participant hourly estimates from the load profiling, aggregation, and loss adjustment processes.

### Imbalances

An imbalance is the difference between two load shapes representing different estimates of the same period, typically a scheduled load shape and an actual load shape. Imbalances are typically used when money has changed hands based on an estimate, which has now been improved or finalized by the settlement process.

### Market Prices

Market prices include the prices for the products and services being billed. These include the hour-by-hour prices for energy within a given market, but also costs for transmission and distribution charges.

### Ancillary Services

Ancillary services are those services necessary to support the transmission of energy from resources to loads while maintaining reliable operation of transmission systems in accordance with good utility practice.

---

## Monthly Settlement Example

A common implementation of Oracle Utilities Load Profiling and Settlement might be to perform monthly settlement. Performing monthly settlement with Oracle Utilities Load Profiling and Settlement involves performing a number of steps each day. These steps are outlined below.

### Step One: Import Data

The first step is to import data to the Oracle Utilities Data Repository.

For customer/account Data, each day a file is generated by the CIS system to update Oracle Utilities Load Profiling and Settlement with records, including:

- Meter reads
- Supplier switches
- Account closings
- New accounts, etc.

Interval Data to be imported includes:

- Interval metered accounts
- System Load
- Service Coordinator (SC) Schedules
- Market Prices
- SIC (System Incremental Cost).

### Step Two: Validate Data

The next step is to validate the imported data. Interval data validation can be performed using the Oracle Utilities Rules Language. Interval data can also be validated through the Validation, Editing, and Estimation (VEE) capabilities of Oracle Utilities Load Analysis (available under separate license).

Customer (relational) data is validated via referential integrity checks during the Import process. Additional validations can be performed using the Oracle Utilities Rules Language.

### Step Three: Calculate Class Profiles for Use in Settlement

The next step is to calculate class profiles that will be used in initial and final settlement. This involves the following steps:

1. Class profiles are calculated using an adjusted static load profiling methodology.
2. The current system load is divided by the historic system load from the same day as the original static profile (1 year, shifted for day of week).
3. The historic class load profile is multiplied by the resulting ratio to arrive at an adjusted class profile.
4. The new adjusted static load profile is stored in the Interval Database for that settlement day/date.

### Step Four: Pre-Aggregate Meter Reads

The next step involves pre-aggregation of meter reads.

All the Bill History records that have the same start and stop dates, same supplier, profile class, loss class, and bus location, and that didn't change any of these relationships during the meter

cycle and have a profile status flag of “No”, are aggregated. This process creates records in the Usage Summary Table that represent the energy of the underlying Bill History records, without requiring the user to profile each record individually.

The number of records in the Usage Summary Table depends on the number of cycles, the number of different start and stop combinations, the number of suppliers, profile classes, bus locations, loss classes, etc. Typically, a million Bill History records will reduce to a few thousand records in the Usage Summary Table.

## Step Five: Profile Available Aggregated Meter Read Records

The next step is to profile the available aggregated meter read records. This involves:

1. Obtaining monthly meter reading and applicable read dates from records in the Usage Summary Table
2. Retrieving class profiles for appropriate dates
3. Multiplying class profiles by the ratio of the monthly meter energy divided by the total energy in the class profile to generate an entity-specific profile
4. Applying hourly loss factors.

## Step Six: Profile Remaining Meter Reads, Estimate for Missing Reads

The next step is to profile the remaining meter read accounts and estimate any missing data. Because all meter reads aren't available by the 15th of the following month, all accounts without data for any day in the month are profiled based on a usage factor. For each day for which an account does not have a profiled Bill History record, the class profile for that day is scaled to the account level using an account scalar factor, and then hourly loss factors are applied.

This process creates a load shape that increases in magnitude as the month progresses. This load shape, when added to the Bill History profiled load shape that decreases as the month progresses, completes the estimate of profiled load during the month.

## Step Seven: Interval Metered Customers

The next step is to process accounts/customers that have interval data recorders. Whenever possible, available interval data is used in the settlement process. For interval metered accounts whose interval data should be used in the settlement process, but is unavailable, the following steps are performed:

1. Create hourly ratio of current system load as a percentage of historical system load
2. Multiply historical customer usage by ratio to arrive at the estimated customer load
3. Apply losses.

## Step Eight: Aggregation

The next step is data aggregation. Individual records are aggregated by Energy Service Provider (ESP), and ESPs are aggregated to Service Coordinators (SCs), as appropriate.

## Step Nine: UFE

The next step is to calculate and allocate Unaccounted for Energy (UFE). The total of all the Service Coordinators' loads is compared to the system load to determine the system Unaccounted for Energy (UFE). Each SC is allocated a portion of UFE by multiplying the total UFE by the ratio of each SC's hourly load to the total of all SCs' hourly loads.

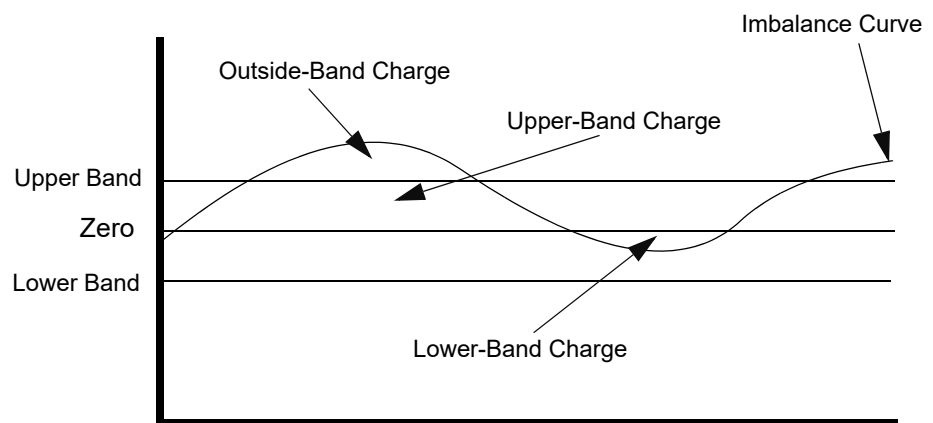
## Step Ten: Imbalances and Financial Settlement

The next step is to calculate any imbalances and perform financial settlement.

The difference between the Service Coordinator's scheduled and estimated usage is calculated as an imbalance. UFE is most often priced based on a System Incremental Cost (SIC), and imbalances have a pricing matrix with a penalty matrix. One approach is to swap energy between the SC's UFE and imbalance in the intervals, where the absolute value of the imbalance is moved closer to zero.

The Net UFE and Net Imbalance are then used in the financial settlement process. A band of the greater of 2 MW or 1.5% of hourly load is applied to the remaining imbalance curve, and energy within the bands is priced at SIC.

For all intervals with an imbalance outside of the band (see Outside-Band Charge in the diagram below), a penalty matrix applies. First, the average absolute value percent outside of the band is calculated for the settlement month. This determines the penalty percentages for each block of hours for which the SC imbalance was outside of the band. The penalty percentage for each hour outside of the band for hours 1 - 100, 101 - 200, 201 - 300, 301 - 400, and 401+ is selected based on the average absolute percent outside the schedule. The net imbalance shape is priced using the matrix penalty amounts.



## Step Eleven: Service Coordinator Billing

The last step is to bill Service Coordinators. Summarized billing determinants and costs for each Service Coordinator for the settlement month are calculated and written to a record in the Bill History Table. Each SC is then billed using Oracle Utilities Billing Component, based on the SC's aggregated billing determinants and costs. In addition, any required reporting is performed at this time.

## Final Settlement

The final settlement is similar to the above sample Monthly Settlement process, except there is no need to perform the usage factor profiling portion because all meter reads are available. The results from the final settlement are compared to the estimated Monthly Settlement in order to calculate the differences and invoice them appropriately.



# Chapter 3

---

## **Oracle Utilities Load Profiling and Settlement Network Environment and Database Installation**

This chapter describes the Oracle Utilities Load Profiling and Settlement network environment and how to install the Oracle Utilities Data Repository used by Oracle Utilities Load Profiling and Settlement, including:

- **The Oracle Utilities Load Profiling and Settlement Installation Package**
- **Oracle Utilities Load Profiling and Settlement Network Environment**
- **Oracle Utilities Load Profiling and Settlement Database Installation**

# The Oracle Utilities Load Profiling and Settlement Installation Package

There is a separate installation package for the Oracle Utilities Energy Information Platform and for each related product. The table below provides the installation package file names for the Oracle Utilities Energy Information Platform and Oracle Utilities Load Profiling and Settlement.

Product	Installation File Name
Oracle Utilities Energy Information Platform	1.6.1.xx.0.EIP.zip
Oracle Utilities Load Profiling and Settlement	1.6.1.xx.0.LPS.zip

The Oracle Utilities Load Profiling and Settlement installation package contains the following folders:

- **Install:** Contains the installation program for Oracle Utilities Load Profiling and Settlement, including:
  - Oracle Utilities LPS 1.6.1.xx.0.msi
- **DBScripts:** Contains the following database scripts for Oracle Utilities Load Profiling and Settlement:
  - Install: scripts used to create a new Load Profiling and Settlement schema
  - Upgrade: scripts used to update the Load Profiling and Settlement schema

See **Oracle Utilities Load Profiling and Settlement Database Installation** on page 3-9 for more information about installing and upgrading the Oracle Utilities Load Profiling and Settlement database schema.

- **Documentation:** Contains documentation for Oracle Utilities Load Profiling and Settlement, including:
  - Oracle Utilities Load Profiling and Settlement Installation and Configuration Guide
  - Oracle Utilities Load Profiling and Settlement User's Guide

**Note:** Oracle Utilities Load Profiling and Settlement documentation can also be downloaded separately.



---

# Oracle Utilities Load Profiling and Settlement Network Environment

This section contains options for hardware and software specifications for implementation of the Oracle Utilities Energy Information Platform and related products, including:

- Oracle Utilities Billing Component
- Oracle Utilities Load Analysis
- Oracle Utilities Load Profiling and Settlement
- Oracle Utilities Meter Data Management
- Oracle Utilities Quotations Management
- Oracle Utilities Rate Management
- Oracle Utilities Transaction Management

The Energy Information Platform application architecture comprises all of the components typically found in any n-tier enterprise architecture, including client machines, servers, and the supporting network infrastructure.

The Oracle Utilities Energy Information Platform uses five primary types of systems components. Suggested hardware specifications and required software for each component are listed below.

Before installing Oracle Utilities software you should consult with Oracle Utilities Consulting. They will make suggestions for your hardware and network architecture based on your business requirements. You should also consult with Oracle Utilities Consulting if your business requirements change, if your database grows significantly, or if you plan on migrating to a different version of Oracle Utilities software.

**Note:** Supported software versions are subject to change. Please refer to the Release Notes shipped with the software for the latest supported versions.

## Client Workstation - C/S

These are workstations used when running the client/server (C/S) versions of Oracle Utilities applications, such as Data Manager.

### Software for Client Workstation - C/S:

- Supported Operating Systems
  - Windows 8.1
  - Windows 10
- Supported Web Browsers
  - Microsoft Internet Explorer 11 with latest security updates
  - Microsoft Edge (Windows 10 only)
- Relational Database Management System (RDBMS) client with applicable ODBC driver and Data Provider for .NET

Note: When using 64-bit operating systems, the 32-bit version of the database client is required.

- Oracle Utilities Energy Information Platform software
- Microsoft .NET Framework 4.5
- Microsoft Visual C++ Redistributable Packages for Visual Studio 2013 (32-bit)
- Microsoft Data Access Component (MDAC) 2.8 or higher
- Microsoft XML Core Services (MSXML) 6.0 or higher

## Client Workstation - Web

These are workstations used when running the web-enabled versions of Oracle Utilities applications only.

### Software for Client Workstation - Web:

- Supported Operating Systems
  - Windows 8.1
  - Windows 10
- Supported Web Browsers
  - Microsoft Internet Explorer 11 with latest security updates
  - Microsoft Edge (Windows 10 only)

**Note:** For users that require access to both client/server and web browser-based applications, please refer to the requirements listed for "Client Workstations - Thick Client."

## Database Server

The database server houses the Oracle Utilities Data Repository. The hardware platform for the database server may be Windows/Intel or UNIX/RISC.

### Software for Database Server:

- Supported Operating Systems
  - Microsoft Windows Server 2012
  - Microsoft Windows Server 2016
  - LINUX (UNIX/Intel)
  - Sun Microsystems Solaris
  - HP-UX
  - IBM AIX (UNIX/RISC) software

Note: Refer to the *Oracle Utilities Energy Information Platform Quick Install Guide* for additional details about supported operating systems.

- RDBMS Server: Oracle

### Database Platforms:

- Oracle 11g, Release 2 (11.2.0.4)
- Oracle 12c, Release 1 (12.1.0.2 or 12.2.0.1)
- Oracle 19c, Release 1 (19.3.0.0.0)

**Note:** Supported database versions and drivers are subject to change. Please refer to the Oracle Release Notes shipped with the software for the latest compatibility matrix.

## Application/Batch Processing Server

Application servers are used when running Windows Services, such as those employed by the Reporting Framework and Adapter components. The application server(s) can also perform batch processes such as meter data upload and validation as well as batch billing and settlements.

### Software for the Application/Batch Server:

- Supported Operating Systems
  - Windows 2012 Server R2
  - Windows 2016 Server
- Supported Web Browsers
  - Microsoft Internet Explorer 11 with latest security updates
  - Microsoft Edge with latest security updates (Windows 2016 Server only)
- RDBMS client with applicable ODBC driver and Data Provider for .NET  
Note: When using 64-bit operating systems, the 32-bit version of the database client is required.
- Oracle Utilities Energy Information Platform software
- Microsoft .NET Framework 4.5
- Microsoft Visual C++ Redistributable Packages for Visual Studio 2013 (32-bit)
- Microsoft Data Access Component (MDAC) 2.8 or higher
- Microsoft XML Core Services (MSXML) 6.0 or higher
- Java Development Kit (JDK) 1.8 or later (for application servers that will be running the Adapter)

## Web Server(s)

Web servers run Microsoft Internet Information Service (IIS). These servers may be combined with the application server(s) for combination Web and Application servers.

### Software for Web Server(s):

- Supported Operating Systems
  - Windows 2012 R2 Enterprise Server with Microsoft Internet Information Service (IIS) 8.5
  - Windows 2016 Enterprise Server with Microsoft Internet Information Service (IIS) 10.0
- Supported Web Browsers
  - Microsoft Internet Explorer 11 with latest security updates
  - Microsoft Edge with latest security updates (Windows 2016 Server only)
- RDBMS client with applicable ODBC driver and Data Provider for .NET

Note: When using 64-bit operating systems, the 32-bit version of the database client is required.
- Oracle Utilities Energy Information Platform software
- Microsoft .NET Framework 4.5
- Microsoft Visual C++ Redistributable Packages for Visual Studio 2013 (32-bit)
- Microsoft Data Access Component (MDAC) 2.8 or higher
- Microsoft XML Core Services (MSXML) 6.0 or higher

# Oracle Utilities Load Profiling and Settlement Database Installation

This section describes how to install and verify the Oracle Utilities Load Profiling and Settlement database tables and data in the Oracle Utilities Data Repository, including:

- **Installation Requirements**
- **Installing the Database**
- **Verifying the Database**

**Note:** This section assumes that you have created the Oracle Utilities Data Repository schema as described in **Chapter 3: Oracle Utilities Data Repository Schema Creation** in the *Oracle Utilities Energy Information Platform Installation Guide*.

## Installation Requirements

The following are required in order to install the Oracle Utilities Load Profiling and Settlement tables and data into the Oracle Utilities Data Repository:

- The Oracle Utilities Data Repository (v1.6.1.0.0) schema must have been installed on the database instance on which you plan to run the Oracle Utilities Load Profiling and Settlement application.
- The **addLPS.cmd** file. This database script adds the Oracle Utilities Load Profiling and Settlement database tables and data to the Oracle Utilities Data Repository.

## Installing the Database

Open a command prompt and run the addLPS.CMD script. This script uses the following syntax:

```
addLPS [-d <database>] [-own <owner name>] -opw <owner password>
```

Parameter	Description
<database>	The name given to the instance as specified in the TNSNAMES.ORA file. This parameter is optional and if not specified, the script will connect to the default Oracle database.
<owner name>	The name of the user which will own database objects. This parameter is optional. If not specified, the default user PWRLINE will own database objects.
<owner password>	The chosen password for the PWRLINE schema owner.

Like the base schema database tables and indexes, the Oracle Utilities Load Profiling and Settlement objects are created in the default tablespace of the PWRLINE user with default sizing parameters. If these defaults are required to be changed then the scripts may be edited.

## Verifying the Database

To verify that the Oracle Utilities Load Profiling and Settlement tables are in place, use the following procedure:

1. Log into the database using the PWRLINE user (Password =password).
2. Verify that the following tables exist in the database:
  - .Bill History Max Values B View
  - .Bill History Max Values View
  - .Capacity Staging
  - .EH Accounts
  - .IDR Overlaps
  - .IDR Segment
  - .IDR Staging
  - .Output Channel Cut Data
  - .Output Channel Cut Header
  - .Output Channel Cut Validation Messages
  - .Overlaps
  - .Procedure Dates
  - .Profile Channel Cut Data
  - .Profile Channel Cut Header
  - .Profile Channel Cut Validation Messages
  - .Scalar Staging
  - .Track Segment
  - Account Capacity History
  - Account Service History
  - Aggregation Channel Cut Data
  - Aggregation Channel Cut Header
  - Aggregation Channel Cut Validation Messages
  - Aggregation Channel History
  - Aggregation Group
  - Aggregation Group History
  - Aggregation Type
  - Distribution Company
  - Loss Class
  - Market
  - Method
  - Process History
  - Profile Class
  - REP



- REP Service History
- Result
- Scheduling Coordinator
- Service
- Segment Channel History
- Settlement Type
- Superzone
- Zone
- Zone Service History

**Appendix A: Oracle Utilities Data Repository Load Profiling and Settlement Database Schema** includes a diagram of the Oracle Utilities Load Profiling and Settlement database schema (v1.6.0.0.0) that provides details regarding the table and columns in the Oracle Utilities Load Profiling and Settlement schema, as well as the relationships between these tables in the Oracle Utilities Data Repository.



# Chapter 4

---

## Installing the Oracle Utilities Load Profiling and Settlement Application Software

This chapter describes how you install the Oracle Utilities Load Profiling and Settlement application software, including:

- **Installing the Oracle Utilities Load Profiling and Settlement Software**
- **Setting Up Configuration Files on the Web Server**
- **Setting Up Configuration Files on Application Servers and Workstations**
- **Installing Oracle Utilities Load Profiling and Settlement Database Records and Rate Forms**

## Installing the Oracle Utilities Load Profiling and Settlement Software

This section describes how to install the Oracle Utilities Load Profiling and Settlement software including:

- **Installing in Conjunction with the Energy Information Platform**
- **Installing After Installation of the Energy Information Platform**

### Installing in Conjunction with the Energy Information Platform

To install the Oracle Utilities Load Profiling and Settlement software in conjunction with the Oracle Utilities Energy Information Platform, use the following procedure:

1. Navigate to the Install folder created by the Energy Information Platform installation package (1.6.1.xx.0.EIP.zip). This folder contains the following files:
  - Oracle Utilities EIP 1.6.1.xx.0.msi
  - setup.exe
2. Navigate to the Install folder created by the Oracle Utilities Load Profiling and Settlement installation package (01.06.10.xx.00.LPS.zip). This folder contains the following:
  - Oracle Utilities LPS 1.6.1.xx.0.msi
3. Move the "Oracle Utilities LPS 01.06.10.xx.00.msi" file into the same directory as the Energy Information Platform files.

The directory should now contain following:

- Oracle Utilities EIP 1.6.1.xx.0.msi
  - setup.exe
  - Oracle Utilities LPS 1.6.1.xx.0.msi
4. Double-click the setup.exe file.

A dialog opens asking you to confirm the products you wish to install. Click **Yes** to continue with the installation. Click **No** to cancel the installation.
  5. Proceed with the installation (starting at Step 3) as outlined in **Chapter 4: Installing the Oracle Utilities Application Software** in the *Oracle Utilities Energy Information Platform Installation Guide*.

### Installing After Installation of the Energy Information Platform

To install the Oracle Utilities Load Profiling and Settlement software after the Oracle Utilities Energy Information Platform software has been installed, use the following procedure:

1. Navigate to the Install folder created by the Oracle Utilities Load Profiling and Settlement installation package (1.6.1.xx.0.LPS.zip). This folder contains the following:
  - Oracle Utilities LPS 1.6.1.xx.0.msi
2. Double-click the Oracle Utilities LPS 1.6.1.xx.0.msi file.

A dialog opens asking you to confirm the products you wish to install. Click **Yes** to continue with the installation. Click **No** to cancel the installation.

3. Proceed with the installation (starting at Step 3) as outlined in **Chapter 4: Installing the Oracle Utilities Application Software** in the *Oracle Utilities Energy Information Platform Installation Guide*.

---

# Setting Up Configuration Files on the Web Server

Oracle Utilities web applications use the following configuration files.

## LODESTAR.CFG

The LODESTAR.CFG file is a text file used to customize the working environment of the Oracle Utilities application software. See **LODESTAR.CFG** on page 2-2 in the *Oracle Utilities Energy Information Platform Configuration Guide* for more information about setting up this file. A sample of this file can be found in the **C:\LODESTAR\CFG\Examples\CFG** directory on the web server.

**Note:** This file **MUST** be named LODESTAR.CFG on the web server.

## LSSECURE.CFG.XML

The LSSECURE.CFG.XML file specifies the data source used by the Security functionality and is required in order to run Oracle Utilities web-enabled applications. The LSSECURE.CFG.XML must be installed in the **C:\LODESTAR\CFG** directory. See **LSSECURE.CFG.XML** on page 2-42 in the *Oracle Utilities Energy Information Platform Configuration Guide* for more information about setting up this file. A sample of this file can be found in the **C:\LODESTAR\CFG\Examples\CFG** directory on the web server.

**Note:** If an application server and web server are installed on different machines, the LSSECURE.CFG.XML file is installed in the **C:\LODESTAR\CFG** directory with the web server components.

## LSREPORTMONITOR.CFG.XML (optional)

The LSREPORTMONITOR.CFG.XML file specifies where report data is stored and how reports are processed through the web-enabled Oracle Utilities Energy Information Platform. The LSREPORTMONITOR.CFG.XML must be installed in the **C:\LODESTAR\CFG** directory. See **LSREPORTMONITOR.CFG.XML** on page 2-36 in the *Oracle Utilities Energy Information Platform Configuration Guide* for more information about setting up this file. A sample of this file can be found in the **C:\LODESTAR\CFG\Examples\CFG** directory on the web server.

**Note:** If an application server and web server are installed on different machines, the LSREPORTMONITOR.CFG.XML file is installed in the **C:\LODESTAR\CFG** directory on both server.

## LSLOGGER.CFG.XML (optional)

The LSLOGGER.CFG.XML file defines how log files are generated by Oracle Utilities applications. Each Oracle Utilities component can be configured to write messages to a specified log file or an Event Log. The LSLOGGER.CFG.XML must be installed in the **C:\LODESTAR\CFG** directory. See **LSLOGGER.CFG.XML** on page 2-27 in the *Oracle Utilities Energy Information Platform Configuration Guide* for more information about setting up this file. A sample of this file can be found in the **C:\LODESTAR\CFG\Examples\CFG** directory on the web server.

**Note:** If an application server and web server are installed on different machines, the LSLOGGER.CFG.XML file is installed in the **C:\LODESTAR\CFG** directory on both machines.

## LSSCHDLR.CFG.XML (optional)

The LSSCHDLR.CFG.XML file defines the data source(s) monitored by the Oracle Utilities Schedule Service (LSSCHDLR.exe). This service monitors the Scheduled Message (SCHEDULEDMESSAGE) table in each data source specified in this file. This file is required if the Oracle Utilities Schedule Service (LSSCHDLR.exe) is used. The LSSCHDLR.CFG.XML must

be installed in the **C:\LODESTAR\CFG** directory. See **LSSCHDLR.CFG.XML** on page 2-41 in the *Oracle Utilities Energy Information Platform Configuration Guide* for more information about setting up this file. A sample of this file can be found in the **C:\LODESTAR\CFG\Examples\CFG** directory on the web server.

**Note:** If an application server and web server are installed on different machines, the LSSCHDLR.CFG.XML file is installed in the **C:\LODESTAR\CFG** directory on both machines.

### **LSRELAY.CFG.XML (optional)**

The LSRELAY.CFG.XML file identifies the machine running an SMTP service used to send email messages from the Oracle Utilities Energy Information Platform. This file is required only if the email functions are used. The LSRELAY.CFG.XML file must be installed in the **C:\LODESTAR\CFG** directory. See **LSRELAY.CFG.XML** on page 2-34 in the *Oracle Utilities Energy Information Platform Configuration Guide* for more information about setting up this file. A sample of this file can be found in the **C:\LODESTAR\CFG\Examples\CFG** directory on the web server.

### **LOCALES.CFG.XML**

The LOCALES.CFG.XML file specifies the regional locales available to Oracle Utilities web-enabled applications. The LOCALES.CFG.XML file must be installed in the **C:\LODESTAR\Web\CCS** directory. A sample of this file can be found in the **C:\LODESTAR\CFG\Examples\CFG** directory on the web server. See **LOCALES.CFG.XML** on page 2-22 in the *Oracle Utilities Energy Information Platform Configuration Guide* for more information about this file.

**Note:** If an application server and web server are installed on different machines, the LOCALES.CFG.XML file is installed in the **C:\LODESTAR\Web\ccs** directory with the web server components.

# Setting Up Configuration Files on Application Servers and Workstations

You also need to set up the configuration files needed by application server and workstation applications such as Data Manager and batch executables.

## **LODESTAR.CFG**

The **LODESTAR.CFG** (also called **POWRLINE.CFG** in older installations) file is a text file used to customize the working environment of the Oracle Utilities application software. See **LODESTAR.CFG** on page 2-2 of the *Oracle Utilities Energy Information Platform Configuration Guide* for more information about setting up this file.

## **LSRELAY.CFG.XML (optional)**

The **LSRELAY.CFG.XML** file identifies the machine running an SMTP service used to send email messages from the Oracle Utilities Energy Information Platform. This file is required only if the email functions are used. The **LSRELAY.CFG.XML** file must be installed in the **C:\LODESTAR\CFG** directory on any application server or workstation used to process Rules Language configuration that uses the **EMAILCLIENT** function. See **LSRELAY.CFG.XML** on page 2-34 in the *Oracle Utilities Energy Information Platform Configuration Guide* for more information about setting up this file. A sample of this file can be found in the **C:\LODESTAR\CFG\Examples\CFG** directory on the web server.

## Installing Oracle Utilities Load Profiling and Settlement Database Records and Rate Forms

The web-enabled Oracle Utilities Load Profiling and Settlement application requires a specific set of database records and Rules Language rate forms be in place in the Oracle Utilities Data Repository. These records are installed in the

**C:\LODESTAR\Web\settlement\rateschedules** directory on the web server where the Oracle Utilities Load Profiling and Settlement web application was installed, and include the following:

### Files

- Acct\_Data.imp
- profiles.lse

### Rate Schedules

- LSLP\_SETTLEMENT\_CONTROL\_SCHEDULE.prg
- LSLP\_SETTLEMENT\_REAGG\_SCHEDULE.prg

### Riders

- LSLP\_AGGREGATION\_CALCS\_RDR.prg
- LSLP\_SAVE\_PROCESSHISTORY\_RDR.prg
- LSLP\_SETTLEMENT\_CALCS\_RDR.prg
- LSLP\_SETTLEMENT\_SAVECHAN\_OVERLAP\_RDR.prg
- LSLP\_SETTLEMENT\_SET\_DEFAULT\_RDR.prg
- LSLP\_SETTLEMENT\_VIA\_DYNGUI\_RDR.prg

### Lists

- LOSS\_LIST.qry
- LSLP\_GET\_MAX\_CHANNEL\_FOR\_RECORDER.qry
- LSLP\_GET\_SETCHANNELHIST\_FOR\_AGGR.qry
- PROFILE\_LIST.qry
- RESULT\_LIST.qry
- SC\_ACCOUNTID\_LIST.qry
- SC\_RECORDS\_LIST.qry
- ZONE\_LIST.qry

These records are installed with the Oracle Utilities Load Profiling and Settlement database schema. These can also be installed manually using either Data Manager or the PLIMPORT, INTDIMP, RFIMPEXP, and LSTIMEXP command line programs. See the *Data Manager User's Guide* and the *Oracle Utilities Energy Information Platform Configuration Guide* for more information about using these tools.



# Chapter 5

---

## Setting Up Oracle Utilities Load Profiling and Settlement Database Records

This chapter describes how to set up database records used when using Oracle Utilities Load Profiling and Settlement, including:

- **Setting Up Oracle Utilities Load Profiling and Settlement Lookup Tables**

You set up database records in the Oracle Utilities Data Repository using either Data Manager or Data Navigator.

## Setting Up Oracle Utilities Load Profiling and Settlement Lookup Tables

The Oracle Utilities Load Profiling and Settlement functions require that records be set up in the following tables in the Oracle Utilities Data Repository:

- **Aggregation Type Table**
- **Distribution Company Table**
- **Loss Class Table**
- **Market Table**
- **Method Table**
- **Profile Class Table**
- **REP Table**
- **Scheduling Coordinator Table**
- **Service Table**
- **Settlement Type Table**
- **Superzone Table**
- **Zone Table**

### Aggregation Type Table

Records in the Aggregation Type table represent types of aggregation (by Loss Class, by Profile Class, by Zone, etc.) that can be performed. Records in this table include the following information:

- **Aggregation Type Code:** A unique code for the aggregation type.
- **Aggregation Type Name:** The name of the aggregation type.
- **Start Time:** The start time for the aggregation type.
- **Stop Time:** The stop time for the aggregation type.

### Distribution Company Table

Records in the Distribution Company table represent distribution companies within the market(s) being settled. Records in this table include the following information:

- **DISCO Code:** A unique code for the distribution company.
- **DISCO Name:** The name of the distribution company.
- **Start Time:** The start time for the distribution company.
- **Stop Time:** The stop time for the distribution company.

### Loss Class Table

Records in the Loss Class table represent individual loss classes used in settlement calculations. Records in this table include the following information:

- **Loss Code:** A unique code for the loss class.
- **Loss Name:** The name of the loss class.
- **Start Time:** The start time for the loss class.

- **Stop Time:** The stop time for the loss class.

## Market Table

Records in the Market table represent the market(s) being settled. Records in this table include the following information:

- **Market Code:** A unique code for the market.
- **Market Name:** The name of the market.
- **Start Time:** The start time for the market.
- **Stop Time:** The stop time for the market.

## Method Table

Records in the Method table represent settlement methods (Estimated History, Estimated No History, Usage Summary, Usage Remainder, etc.) that can be used in settlement calculations. Records in this table include the following information:

- **Method Code:** A unique code for the method.
- **Method Name:** The name of the method.
- **Start Time:** The start time for the method.
- **Stop Time:** The stop time for the method.

## Profile Class Table

Records in the Profile Class table represent individual profile classes (commercial, residential, etc.) used in settlement calculations. Records in this table include the following information:

- **Profile Code:** A unique code for the profile class.
- **Profile Name:** The name of the profile class.
- **Start Time:** The start time for the profile class.
- **Stop Time:** The stop time for the profile class.

## REP Table

Records in the REP table represent retail energy providers (REPs) within the market(s) being settled. Records in this table include the following information:

- **REP Code:** A unique code for the REP.
- **REP Name:** The name of the REP.
- **Start Time:** The start time for the REP.
- **Stop Time:** The stop time for the REP.

## Scheduling Coordinator Table

Records in the Scheduling Coordinator table represent scheduling coordinators within the market(s) being settled. Records in this table include the following information:

- **Scheduling Coordinator Code:** A unique code for the scheduling coordinator.
- **Scheduling Coordinator Name:** The name of the scheduling coordinator.
- **Start Time:** The start time for the scheduling coordinator.

- **Stop Time:** The stop time for the scheduling coordinator.

## Service Table

Records in the Service table represent individual services (electric, gas, etc.) for settlement calculations are being performed. Records in this table include the following information:

- **Code:** A unique code for the service.
- **Name:** The name of the service.
- **Unit of Measure:** The optional unit of measure (from the UOM Table) associated with the service.
- **End Use:** The optional end use (from the End Use table) associated with the service.

## Settlement Type Table

Records in the Settlement Type table represent types of settlements (initial, final, true up) that can be performed. Records in this table include the following information:

- **Settlement Type Code:** A unique code for the settlement type.
- **Settlement Type Name:** The name of the settlement type.

## Superzone Table

Records in the Superzone table represent the superzone(s) within the market(s) being settled. Records in this table include the following information:

- **Superzone Code:** A unique code for the superzone.
- **Superzone Name:** The name of the superzone.
- **Start Time:** The start time for the superzone.
- **Stop Time:** The stop time for the superzone.

## Zone Table

Records in the Zone table represent the zone(s) with the market(s) being settled. Records in this table include the following information:

- **Zone Code:** A unique code for the zone.
- **Zone Name:** The name of the zone.
- **Start Time:** The start time for the zone.
- **Stop Time:** The stop time for the zone.

# Chapter 6

---

## Configuring Oracle Utilities Load Profiling and Settlement

This chapter describes how to configure Oracle Utilities Load Profiling and Settlement, including adding customized business logic to Oracle Utilities Load Profiling and Settlement, including:

- **Customizing Oracle Utilities Load Profiling and Settlement Processes**
- **Customizing Parameters for Settlement Processing**

## Customizing Oracle Utilities Load Profiling and Settlement Processes

This section describes how to customize the settlement and aggregation processes run by Oracle Utilities Load Profiling and Settlement.

Oracle Utilities Load Profiling and Settlement performs a number of processes via pre-configured Rules Language rate schedules and riders. These processes include:

- Settlement Calculations (LSLP\_SETTLEMENT\_CONTROL\_SCHEDULE)
- Aggregation Calculations (LSLP\_SETTLEMENT\_REAGG\_SCHEDULE)

These rate schedules include riders (supplied by Oracle Utilities) that can be used to add customized logic to that particular process using the Rules Language. These riders include:

- Settlement Calculations (LSLP\_SETTLEMENT\_CALCCS\_RDR)
- Aggregation Calculations (LSLP\_AGGREGATION\_CALCCS\_RDR)

### Adding Custom Logic to Oracle Utilities Load Profiling and Settlement Processes

To add customized business logic to one or more of the above Oracle Utilities Load Profiling and Settlement processes, either edit the appropriate supplied rider, or create a new version of the rider, and add appropriate Rules Language configuration. See **How to open an existing rate form version:** on page 2-3 in the *Oracle Utilities Rules Language User's Guide* for more information about opening existing rate form versions. See **How to open a new rate form version:** on page 2-2 in the *Oracle Utilities Rules Language User's Guide* for more information about creating new rate form versions.

For example, to add customized logic to the Aggregation Calculation process, you could edit the LSLP\_AGGREGATION\_CALCCS\_RDR and include the custom Rules Language configuration to perform the desired logic.

## Customizing Parameters for Settlement Processing

When installed, Oracle Utilities Load Profiling and Settlement performs settlement and aggregation processing based on pre-defined rate schedules and riders defined in the Oracle Utilities Rules Language. These processes use a default set of parameters that include factors such as Market Code, Settlement Type, Start Time, Stop Time, and others. However if needed, these processes can use custom in addition to (or in the place of) the default parameters.

Customizing parameters for Oracle Utilities Load Profiling and Settlement web calculations involves two steps:

- **Customize Parameters in Rate Schedule XSD**
- **Customize Parameters in Oracle Utilities Load Profiling and Settlement Rules Language**

### Customize Parameters in Rate Schedule XSD

The first step is to customize the parameters as desired in the \*.XSD file that defines the Run Settlement screen. This file is called **SettlementRates.xsd** and is installed in the C:\LODESTAR\Web\settlement directory on the web server where the Oracle Utilities Load Profiling and Settlement web application is installed.

The SettlementRates.xsd as installed is as follows (parameters are in bold face):

```
<xs:schema xmlns:xs=http://www.w3.org/2001/XMLSchema
  xmlns:ls="http://www.lodestarcorp.com" attributeFormDefault="unqualified"
  elementFormDefault="qualified">
  <xs:element name="Call">
    <xs:complexType>
      <xs:all>
        <xs:all>
          <xs:element name="RunSettlement" ls:CallAsync="true">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="MARKETCODE" type="ls:lookup"
                  ls:entity="LSLPMARKET" ls:isRequired="true" />
                <xs:element name="SETTLEMENTTYPECODE" type="ls:lookup"
                  ls:entity="LSLPSETTLEMENTTYPE" ls:isRequired="true" />
                <xs:element name="SETTLEMENTSTARTTIME" type="xs:dateTime"
                  ls:isRequired="true" />
                <xs:element name="SETTLEMENTSTOPTIME" type="xs:dateTime"
                  ls:isRequired="true" />
                <xs:element name="TARGETDATE" type="xs:dateTime"
                  ls:isRequired="true"/>
                <xs:element name="LASTDATE" type="xs:date"
                  ls:isRequired="true" />
              </xs:sequence>
              <xs:attribute name="RATE_CODE"
                fixed="LSLP_SETTLEMENT_CONTROL_SCHEDULE" />
              <xs:attribute name="OPERATING_COMPANY" fixed="OPCO" />
              <xs:attribute name="JURISDICTION" fixed="JURIS" />
            </xs:complexType>
          </xs:element>
          <xs:element name="RunAggregation">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="MARKETCODE" type="xs:string"/>
                <xs:element name="SETTLEMENTTYPECODE" type="xs:string"/>
                <xs:element name="SETTLEMENTSTARTTIME" type="xs:dateTime"/>
                <xs:element name="SETTLEMENTSTOPTIME" type="xs:dateTime"/>
                <xs:element name="SAVECHAN" type="xs:string"/>
                <xs:element name="AGGREGATIONS" type="xs:string"/>
              </xs:sequence>
              <xs:attribute name="RATE_CODE"
                fixed="LSLP_SETTLEMENT_REAGG_SCHEDULE" />
              <xs:attribute name="OPERATING_COMPANY" fixed="OPCO" />
            </xs:complexType>
          </xs:element>
        </xs:all>
      </xs:complexType>
    </xs:element>
  </xs:schema>
```

```

        <xs:attribute name="JURISDICTION" fixed="JURIS" />
    </xs:complexType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>
</xs:schema>

```

## Parameter Format

Parameters in this file use the following format:

```

<xs:element name=<PARAMETER> type="ls:lookup" ls:entity=<TABLE_NAME>
ls:isRequired="true" />

```

where:

- <PARAMETER> is the name of the parameter.
- **Type:** The type of parameter. Can be one of the following:
  - **Date** (xs:date): Defines a date parameter. Displays a Date control on the user interface.
  - **Datetime** (xs:dateTime): Defines a datetime parameter. Displays a DateTime control on the user interface.
  - **Lookup** (ls:lookup): Defines a lookup parameter. Displays a Lookup control on the user interface.
  - **Value** (xs:string): Defines a string parameter. Displays a text field on the user interface.
- <TABLE\_NAME> (for Lookups only) is the name of the table in the Oracle Utilities Data Repository that is used for the look up.
- **Required:** Specifies whether or not the parameter is required.
  - **Required:** ls:isRequired="true"
  - **Optional:** ls:isRequired="false"

## Working with parameters

You can add new parameters, edit or change existing parameters, or remove parameters.

- To change parameters used by Settlement calculations, change parameters within the `<xs:element name="RunSettlement" ls:CallAsync="true">` element.
- To change parameters used by Aggregation calculations, change parameters within the `<xs:element name="RunAggregation">` element.

## Adding Parameters

To add a parameter, add an `<xs:element...>` element that defines the parameter you wish to add to the SettlementRates.xsd file. You also need to add a line to the **dictionary.dic** file in the C:\LODESTAR\web\settlement directory that defines the label for the new parameter.

For example, to add an optional parameter called Report Type (from a custom table called REPORTTYPE), you would add the following line to the SettlementRates.xsd file

```

<xs:element name="REPORTTYPE" type="ls:lookup" ls:entity="LSLPPREPORTTYPE"
ls:isRequired="false" />

```

You would then also add the following line to the **dictionary.dic** file in the C:\LODESTAR\web\settlement directory:

```

<REPORTTYPE ENU="Report Type"/>

```



## Editing Parameters

To edit an existing parameter, edit the <xs:element...> element that corresponds to the parameter you wish to edit from the SettlementRates.xsd file.

To change the label of a parameter on the Run Settlement screen, you only need to change the appropriate entry in the **dictionary.dic** file in the C:\LODESTAR\web\settlement directory.

For example, to change the label of the “Last Date” parameter to “Last Profile Date”, you would change the following line in the **dictionary.dic** file from

```
<LASTDATE ENU="Last Date"/>
```

to

```
<LASTDATE ENU="Last Profile Date"/>
```

## Removing Parameters

To remove a parameter, remove the <xs:element...> element that corresponds to the parameter you wish to remove from the SettlementRates.xsd file.

For example, to remove the “Target Date” parameter, delete the <xs:element name="TARGETDATE"... > element from the SettlementRates.xsd file.

Important Note: Making changes to any other portions of the file can result in errors when using the application.

## Customize Parameters in Oracle Utilities Load Profiling and Settlement Rules Language

After changing the parameters in the SettlementRates.xsd file, make corresponding changes in the rate schedules used in performing the calculations. This involves two steps:

- **Add, Edit, or Delete Parameters**
- **Incorporate Customized Parameters in Rules Language Processing**

### Add, Edit, or Delete Parameters

The parameters used by the Rules Language processing are set in a section of the rate schedules labeled “Set Defaults”, as shown below:

```
//***** Set Defaults Start *****
MARKET = MARKETCODE;
SETTLEMENT_TYPE = SETTLEMENTTYPECODE;
SETTLEMENT_START = SETTLEMENTSTARTTIME;
SETTLEMENT_STOP = SETTLEMENTSTOPTIME;
...
```

To change a parameter, add, edit, or delete the line the rate schedule that corresponds to the updated parameter as appropriate.

- When changing parameters used by Settlement calculations, change parameters within the LSLP\_SETTLEMENT\_CONTROL\_SCHEDULE rate schedule.
- When changing parameters used by Aggregation calculations, change parameters within the LSLP\_SETTLEMENT\_REAGG\_SCHEDULE rate schedule.

For example, to add the optional parameter called Report Type (from a custom table called REPORTTYPE) from the example above, you would add the following line to the LSLP\_SETTLEMENT\_CONTROL\_SCHEDULE rate schedule:

```
REPORT_TYPE = LSLPREPORTTYPE;
```

### Incorporate Customized Parameters in Rules Language Processing

After the customized parameter have been updated in the “Set Defaults” section of the appropriate rate schedule, the last step is to incorporate the customized parameters in the settlement and/or aggregation calculations.

# Chapter 7

---

## Using the Proxy Day Extension with Oracle Utilities Load Profiling and Settlement

This chapter describes the Proxy Day DLL Extension used with Interval Data stored in the Oracle Utilities Data Repository including:

- **How the Proxy Day Extension DLL Works**
- **Required Components**
- **Proxy Day Extension DLL Inputs**
- **Proxy Day Return Codes**
- **Proxy Day Tests and DST**

## How the Proxy Day Extension DLL Works

The Proxy Day Extension DLL allows users to employ the functionality of the Proxy Day Selection Program (a component of Oracle Utilities Load Analysis) using the Oracle Utilities Rules Language. The Proxy Day Selection Program selects a historical day (proxy Day) whose load shape most accurately represents a supplied day's load shape. The Proxy Day Extension DLL allows proxy day selection routines to be encoded into rate or settlement schedules and the selected dates to be used in the profile creation and application processes without the need for intervention.

The Proxy Day Extension DLL is triggered using the "USEREXIT" Rules Language function. An example rate schedule that triggers the Proxy Day Extension DLL is provided later in this document.

### Some Important Notes

- Use of Proxy Day DLL with a relational database requires updates to the Oracle Utilities Data Repository including the addition of a number of new tables that store interval data in an appropriate format. These tables include the following:
  - LSCHALDB (stores files in ALDB format)
  - LSCHCLDB (stores files in CLDB format)
  - LSCHELDB (stores files in ELDB format)
  - LSCHSLDB (stores files in SLDB format)
  - LSCHGLDB (stores files in GLDB format)
  - LSCHRLDB (stores files in RLDB format)
  - LSCHCLDB2 (stores temperature data in CLDB format)
  - LSCHGLDB2 (stores temperature data in GLDB format)

**Note:** The GLDB format is not supported by the Proxy Day DLL when using the Oracle Utilities Data Repository.

In addition, there are a number of tables related to each of the above tables that store related header information, edit trail information, and validation information.

You cannot currently import data into these tables using the Interval Data Import Utility in Data Manager. To import data into these tables, use the RDB/LSCHCLDB format for specifying the target database file in the INTDIMP command line program. For example:

```
intdimp -i c:\powrline\user\proxy_rdb1.lse -o RDB/LSCHCLDB
-f c:\powrline\setup\peipli.cfg -u alapoint -p anneums -d peipli
-l c:\temp\intdimp.log
```

- The RUNRS command line also needs to specify "RDB" in order to access interval data in the Oracle Utilities Data Repository. If the user's **Tools->Options** is set to **Customer Relational Database**, the "-b filename" qualifier is optional. If you decide to use it, it should be specified as "-b RDB".
- You can currently view the interval data once it has been imported using Data Manager using the **Select Interval Data** option (select **File->Browse->Select Interval Data** to access this option).
- The Proxy Day Extension DLL returns a single return-code followed by a comma, and a semicolon-separated list of selected dates in the same string. Reports are not normally produced when the DLL is invoked. See **Data Returned from Proxy Day** on page 7-12 for more information about the data returned from the Proxy Day Extension DLL.

- The Proxy Day Extension DLL uses the LSCalendar.XML configuration file to determine the Daylight Savings Time (DST) rules for a specific cut, based on the **Time Zone Standard Name** value of the cut (if provided). Regions, time zones, and DST rules can be configured in this file. See **LSCALENDAR.CFG.XML** on page 2-23 of the *Oracle Utilities Energy Information Platform Configuration Guide* for more information. See **Time Zone Standard Name** on page B-9 in the *Data Manager User's Guide* for more information about the Time Zone Standard Name field in the Oracle Utilities Enhanced Input/Output (\*.LSE) format.

## Required Components

The following is a list of files that are necessary for the Proxy Day Extension DLL to function properly. These files must be present in the C:\LODESTAR\BIN directory of the Oracle Utilities Load Profiling and Settlement installation.

- Proxyday.dll
- Ibmws20.dll
- Proxday.dll

## Proxy Day Extension DLL Inputs

The Proxy Day Extension DLL requires specific input data in order to function properly. This data must be passed to the DLL via the Oracle Utilities Rules Language, and includes:

- An Environment File,
- A Season File,
- A Holiday File,
- A path to the interval database from which the proxy day is to be selected, and
- A path to an optional temperature database.

This section describes how to create the inputs required by the Proxy Day Extension DLL using the Oracle Utilities Rules Language, which can in turn be used by the **USEREXIT** Rules Language function (described in the *Oracle Utilities Rules Language User's Guide*).

### Environment File

An **Environment File** is a file that contains a set of commands and parameters that define how the program will process the data. Each data element in the environment file should be part of a string that is passed to the Proxy Day Extension DLL. The format for each element is provided below. The environment file used by the Proxy Day Extension DLL contains the following elements:

- **DATE** — *(Required)*. The DATE element is a user-specific date range that the day selected will be within. The minimum date range is one month; the maximum is 36 months. The DATE format is mm/dd/yy. The start time is at 00:01 and the stop time is at 24:00 (full days); any other hours and minutes specified will be ignored.

**Rules Language Format:** "DATE 041682 053182;"

- **KEY** — *(Required)*. The KEY element contains a recorder, channel id and a date that indicates to the program the supplied load shape day to be matched in the selection process. A full day is used; any hours and minutes other than 00:01 will be ignored.

**Rules Language Format:** "KEY N1727, 1, 050182;"

- **PROxy** — *(Optional)*. The PROxy element contains a recorder, channel id that indicates to the program the proxy cut series to search. If this command is omitted, the cut series of the supplied load shape day will be used.

**Rules Language Format:** "PRO N1736, 1;"

- **DAY** — *(Optional)*. The DAY element specifies whether the proxy day selected must be of the same day type as the supplied load shape day in the following criteria: Day of Week, Weekend/Weekday, Any (no daytype test). Possible values for this element include:

- **DT** — Proxy day matches day type (weekday or weekend)
- **DOW** — Day of week matches exactly (mon=mon)
- **ANY** — Any day type qualifies (no checking). ANY is the default.

**Rules Language Format:** "DAY ANY;"

- **HOLIDay checking** — *(Optional)*. If the HOL element is entered, holiday checking will be performed. This means the program may select only potential proxy days with holiday status matching that of the user-supplied day. Up to 100 holidays may be defined over the selection period in the Holiday File (see below), which is required if the HOL element is specified. If this element is not specified, no holiday checking will be performed (holidays will be treated the same as non-holidays in terms of eligibility), and the Holiday File will not be used.

**Rules Language Format:** "HOL;"

- **PEAk separation** — *(Optional)*. If the PEAk separation element is included, the proxy day selected must have its peak time within a user-specified number of minutes of the peak time in the supplied load shape day. If this command is not specified or is specified without a value, the default is 60 minutes. Possible values for this element include:

- **m** — The minutes allowed between the proxy day peak time of day and that of the supplied day.
- **60** — The default.

**Rules Language Format:** "PEA 240 ;"

- **SEASon checking** — *(Optional)*. If the SEA element is entered with a Season Schedule number, season checking will be performed. This means the proxy day selected must be in the same season as the supplied load shape day. The seasons are defined in the Season File (see below). If this command is not entered, no season checking is performed, and the Season File is not accessed.

**Rules Language Format:** "SEA 1 ;"

- **TEMP checking** — *(Optional)*. There are two separate commands for temperature checking (TEM and DEG). If one is specified, both must be specified. The TEMP element may specify either one or two temperature cut series. Possible values for this element include:

- **skey** — skey is the key (recorder,channel) of the temperature cut series providing temperatures for the supplied load shape day.
- **pkey** — pkey is the key of the cut series providing temperatures for the potential proxy day series. If this key is omitted, the skey series will be used to find temperatures for both the supplied load shape day and the potential proxy day series.

**Rules Language Format:** "TEM N1768, 1 ;"

- **DEG** — This element must be specified if the TEMP element is used. Values for this element include:

- **#degrees** — This indicates the number of degrees by which a potential proxy day's temperature may differ from the supplied load shape day's. DEG must be a non-negative integer.
- **AVG or MAX** — This defines which test is to be performed. If AVG is selected, the average temperatures of the potential proxy day and the supplied load shape day will be compared; if MAX is selected, their maximum temperatures will be compared. MAX is the default.
- **MIS #hours** — is the maximum number of hours in the temperature data for any one day that may be missing (status '9') and the temperature eligibility test still be performed. This must be a whole number from 0 to 23; the default is 23 if this number is not specified (i.e., only if all 24 hours of any day's temperature data are missing will temperature testing be bypassed).

**Rules Language Format:** "DEG 10 MAX MIS 4 ;"

- **NUM Reported** — *(Optional)*. The NUM element specifies how many proxy days will be reported. This is provided as an integer between 1 and 20. The default is 1. The program will report up to the top 20 ranked proxy days with their associated ranking values. However, no more days will be reported than meet the specified eligibility criteria.

**Rules Language Format:** "NUM 20 ;"

- **MAGnitude weight** — *(Optional)*. This element indicates to the program what weight to assign to the magnitude test. The magnitude test compares similar interval values in energy units and sums the squared differences between the corresponding hours of the supplied day and the potential proxy day. This element is provided as 1, 0, or a decimal fraction



representing a number of tenths, such as .3 or 0.3. The results of this test can be impacted by DST. See **Proxy Day Tests and DST** for more information.

**Rules Language Format:** "MAG 0.5;"

- **SHApe weight** — *(Optional)*. This command indicates to the program what weight to assign to the shape test. The shape test calculates the incremental load over the previous hour and sum the squared differences between the corresponding values of the supplied day and the potential proxy day. This element is provided as 1, 0, or a decimal fraction representing a number of tenths.

You may specify which tests are to be performed and give a weight to each test. Precision is in tenths. If a weight value is not entered, 1 is assumed. If only one test is selected and no weight specified, the weight value for that test is defaulted to 1 and the other test is not performed. If only one test is selected and a weight is specified, the weight value for the other test is defaulted to (1 – the specified weight). If both tests are selected, the sum of the two weights must equal 1. If neither of the tests is specified in the Environment File, both tests will be performed with equal weights (.5). The results of this test can be impacted by DST. See **Proxy Day Tests and DST** for more information.

**Rules Language Format:** "SHA 0.5;"

- **INEligible** — *(Optional)*. The INEligible element is used to produce the Ineligible Days Report, which lists each day within the date range that did not pass eligibility criteria, along with a message indicating the eligibility test that it failed.

**Rules Language Format:** "INE;"

The Environment file is passed to the Proxy Day Extension DLL in a string containing the specific environment file commands, separated by semicolons.

**Format:** Keyword Keyvalue; Keyword Keyvalue; Keyword Keyvalue;.....

**Example:** DAT 050100 051200; KEY N1723,1,051000; PROXY N2024 2

## Season File

A **Season File** is a schedule that applies different TOU schedules to seasons throughout the year. The Season file is passed to the Proxy Day Extension DLL in a string containing a full path and file name of the Season file. Season files used by the Proxy Day Extension DLL **must** be in the same format as those created using Oracle Utilities Load Analysis.

**Format:** full path and filename

**Example:** c:\lodestar\user\season.dat

## How to Create the Season File

In the Season file, on peak and off peak hours may be defined differently for summer and winter months, or a third, shoulder period can be defined for summer only.

The Season file consists of one or more season schedules. All records for each season schedule are grouped together, in ascending order by season schedule number. Each record specifies a time period, assigns to it a season, and associates with it a Time-Of-Use schedule. A maximum of 12 distinct seasons may be defined in each season schedule.

Many separate time periods may be defined as belonging to the same season (winter in several different years, for example). Within any season schedule, every time period assigned to the same season must also be associated with the same time-of-use schedule.

Each Season file record has the following format:

<b>season-sched#,</b>	<b>season-number</b>	<b>tou sched#,</b>	<b>start,</b>	<b>stop,</b>	<b>season-name</b>
-----------------------	----------------------	--------------------	---------------	--------------	--------------------

where:

Parameter	Description
season-sched#	is a positive integer identifying the season schedule to which this record belongs.
season-number	is a number from 1 to 12 identifying a season.
tou sched#	is a non-negative integer identifying the time-of-use schedule (defined in the Time-Of-Use File) to be used over this season.
start	is the start time of this season segment, in “MM/DD/YY-hh:mm:ss” format. This must be the beginning of a full day; if a time (hh:mm:ss) other than 00:00:00 is specified, it is ignored and the program proceeds using a time of 00:00:00.
stop	is the stop time of this season segment, in “MM/DD/YY-hh:mm:ss” format. This must be the end of a full day; if a time (hh:mm:ss) other than 23:59:59 is specified, it is ignored and the program proceeds using a time of 23:59:59.
season-name	is a description of the season. A given season-number must be paired with the same season-name throughout a single season-sched#.

**Note**

A Season Schedule must contain season segments covering all times in the period being reported or analyzed, with no gaps and no overlaps. If the program detects a gap or an overlap in the designated Season Schedule, it will produce an error message and stop processing. Also, all Time-of-Use schedules referenced by the designated Season Schedule over the analysis period must be defined in the Time-Of-Use File.

The following example shows a Season File containing two season schedules, one defining two distinct seasons and the other defining four. Either Season Schedule one or Season Schedule two could be used for the period from July through December of 1992, for example, but only Season Schedule two could be used over the same months in 1993.

```

1,1,1,11/15/91-00:00:00,04/20/92-23:59:59,WINTER
1,2,2,04/21/92-00:00:00,06/11/92-23:59:59,SPRING
1,3,3,06/12/92-00:00:00,08/31/92-23:59:59,SUMMER
1,4,4,09/01/92-00:00:00,11/14/92-23:59:59,FALL
1,1,1,11/15/92-00:00:00,04/20/93-23:59:59,WINTER
1,2,2,04/21/93-00:00:00,06/11/93-23:59:59,SPRING
1,3,3,06/12/93-00:00:00,08/31/93-23:59:59,SUMMER
1,4,4,09/01/93-00:00:00,11/14/93-23:59:59,FALL

2,1,1,11/12/91-00:00:00,04/20/92-23:59:59,WINTER
2,2,2,04/21/92-00:00:00,11/11/92-23:59:59,NON-WINTER
2,1,1,11/12/92-00:00:00,04/20/93-23:59:59,WINTER
2,2,2,04/21/93-00:00:00,11/11/93-23:59:59,NON-WINTER
2,1,1,11/12/93-00:00:00,04/20/94-23:59:59,WINTER
    
```

This file contains two different schedules. Look closely at the first schedule. Four distinct seasons are defined and associated with different Time-of-Use schedules. Notice how all times in the analysis period are accounted for.

## Holiday File

A **Holiday File** is a list of all national and local holidays observed in your service territory. The Holiday file is passed to the Proxy Day Extension DLL in a string containing a full path and file name of the Holiday file. Holiday files used by the Proxy Day Extension DLL **must** be in the same format as those created using Oracle Utilities Load Analysis.

**Format:** full path and filename

**Example:** c:\lodestar\user\holiday.dat

## How to Create the Holiday File

The Holiday file is required for analysis and reporting programs, since holidays are normally associated with different load patterns. Typically, this file is created at system setup, and added to each year. You should keep past holidays in the file, because the analysis and reporting programs will require the historical information.

You create this file using the following record format:

mm/dd/yy

Enter one date per line. The length of the file is unlimited.

The following is an example of a Holiday file.

```
01/01/87
02/16/87
04/17/87
05/25/87
07/04/87
09/07/87
11/26/87
12/25/87
01/01/88
02/15/88
05/24/88
07/04/88
09/06/88
11/11/88
11/24/88
12/25/88
01/01/89
01/16/89
02/20/89
05/29/89
07/04/89
09/04/89
11/11/89
11/23/89
12/25/89
```

## Interval Database

The interval database is the database from which the proxy day(s) is selected.

For interval data stored in the Oracle Utilities Data Repository, the interval database is passed to the Proxy Day Extension DLL in a string containing a full path to the Oracle Utilities Data Repository and table name.

**Format:** RDB/tablename

**Example:** RDB/LSCHCLDB2

Valid values include:

- LSCHANNELCUTHEADER
- LSCHALDB
- LSCHCLDB
- LSCHCLDB
- LSCHSLDB
- LSCHSLDB
- LSCHSLDB
- LSCHSLDB
- LSCHRLDB

## Temperature Database

If temperature testing is to be done, the temperature database is passed to the Proxy Day Extension DLL in a string containing a full path and file name of the temperature database (or when using data in the Oracle Utilities Data Repository, a path to the Oracle Utilities Data Repository and table name). This may or may not be the same as the interval database. If no temperature testing is to be done, this string should be null.

**Format:** full path and filename

**Example:** c:\lodestar\user\temp.lse

**Example:** RDB/LSCHCLDB2

## Triggering the Proxy Day Extension DLL with the Rules Language

As previously noted, the Proxy Day Extension DLL is triggered using the “USEREXIT” Rules Language function. The format of the “USEREXIT” function is as follows:

```
<identifier> = USEREXIT(<dll_name>, <function_name>[,<arg1>...]);
```

**Where:**

- <dll\_name> is the name of a Dynamic Link Library (DLL) that contains the function. Do not include the ‘.dll’ extension.
- <function\_name> is the name of a function in the DLL.
- <arg#> are arguments to the function.

When triggering the Proxy Day Extension DLL from the USEREXIT function, these parameters include the <dll\_name>, <function\_name>, and up to five <arguments>, as follows:

- <dll\_name>: “PROXYDAY”
- <function\_name>: “PROXYDAY”
- <arg1>: <name of environment file string>
- <arg2>: <name of season file string>
- <arg3>: <name of holiday file string>
- <arg4>: <name of interval database string>

- <arg5>: *optional* <name of temperature database string>

The following examples demonstrate using the Proxy Day Extension DLL with the Oracle Utilities Rules Language.

### Using Interval Data in RDB

```

/* PROXY DAY DLL EXAMPLE SCHEDULE */
/* */
/* Create Environment File String */
DAT_STR = "DATE 041682 053182;";
KEY_STR = "KEY N1727,1,050182;";
PRO_STR = "PRO N1736,1;";
MAG_STR = "MAG 0.5;";
SHA_STR = "SHA 0.5;";
NUM_STR = "NUM 20;";
PEA_STR = "PEA 240;";
DAY_STR = "DAY ANY;";
INE_STR = "INE;";
SEA_STR = "SEA 1;";
HOL_STR = "HOL;";
ENV_FILE_STR = DAT_STR + KEY_STR + PRO_STR + MAG_STR + SHA_STR +
NUM_STR + PEA_STR + DAY_STR + INE_STR + SEA_STR + HOL_STR;

/* Establish Season and Holiday File Strings */
SEASON_FILE_STR = "D:\PROXY_DATA\Tge31e.inf";
HOLIDAY_FILE_STR = "D:\PROXY_DATA\Tge31c.inf";

/* Establish Database File Strings */
INT_DB_PATH = "RDB/LSCHCLDB";
TEMP_DB_PATH = "RDB/LSCHCLDB2";

/* Trigger Proxy Day Function */
PROXY_CALL = USEREXIT("PROXYDAY" , "PROXYDAY" , ENV_FILE_STR ,
SEASON_FILE_STR , HOLIDAY_FILE_STR , INT_DB_PATH , TEMP_DB_PATH);
FINAL_RESULT = PROXY_CALL;

/* Create label for result and include on report */
LABEL FINAL_RESULT "Proxy Day Result"

```

## Data Returned from Proxy Day

Upon successful completion of a run, the Proxy Day Extension DLL returns a character string containing a return code and a sequence of found proxy days (if any are found) separated by semi-colons.

**Format:** return code,top-proxy date; second-best proxy date; third-best proxy date;.... nth best proxy day

**Examples:**

A good run might produce and return a string similar to:

"00000,05/19/00;05/12/00;05/01/00;05/24/00"

A run with errors would produce a string similar to:

"00099,"

### Important Note

There may be cases in which the return string has a non-zero return code yet returns a list of proxy days. This is usually indicative of temperature tests being bypassed due to data or errors with the temperature test. The Return codes on the following pages that are of this nature are designated with an underline.

## Proxy Day Return Codes

The table below lists the return codes (including error codes) provided by the Proxy Day Extension DLL. Underlined return codes indicate that temperature tests were bypassed due to data or errors with the temperature test. In this case, the return string has a non-zero return code yet returns a list of proxy days.

<b>Code</b>	<b>Error</b>
00000	SUCCESSFUL RUN
<u>00003</u>	UNABLE TO OPEN TEMPERATURE DATABASE. REQUESTED TEMPERATURE TESTING NOT PERFORMED.
<u>00005</u>	INSUFFICIENT TEMPERATURE DATA FOR ONE OR MORE PROSPECTIVE PROXY DAYS. REQUESTED TEMPERATURE TESTING NOT PERFORMED.
<u>00006</u>	UOM FOR PROXY TEMPERATURE CUT SERIES INCOMPATIBLE WITH THAT FOR SUPPLIED LOAD SHAPE DAY. REQUESTED TEMPERATURE TESTING NOT PERFORMED.
<u>00007</u>	NON-TEMPERATURE UOM CODE IN PROXY TEMPERATURE CUT SERIES. REQUESTED TEMPERATURE TESTING NOT PERFORMED.
<u>00008</u>	UOM CODE NOT CONSISTENT THROUGHOUT PROXY TEMPERATURE CUT SERIES. REQUESTED TEMPERATURE TESTING NOT PERFORMED.
00009	PROXY DAY PROGRAM ERROR CONDITION
00010	INVALID START DATE
00011	INVALID STOP DATE
00012	STOP DATE DOES NOT FOLLOW START DATE
00013	SELECTION RANGE MUST BE AT LEAST 1 MONTH LONG.
00014	SELECTION RANGE MUST NOT EXCEED 3 YEARS.
00015	MISSING SUPPLIED LOAD SHAPE DAY KEY
00016	SUPPLIED LOAD SHAPE DAY CUSTOMER ID IS INVALID.
00017	SUPPLIED LOAD SHAPE DAY CHANNEL IS INVALID.
00018	NO DATE IN SUPPLIED LOAD SHAPE DAY KEY.
00019	SUPPLIED LOAD SHAPE DATE IS INVALID.
00020	PROXY SERIES KEY MISSING.
00021	NO CHANNEL SUPPLIED FOR PROXY SERIES KEY.
00022	PROXY SERIES KEY CHANNEL IS INVALID.
00023	PROXY SERIES KEY IS INVALID.
00024	INVALID NUMBER OF PROXY DAYS: MUST BE 1-20.
00025	SEASON SCHEDULE MUST BE A NON-NEGATIVE INTEGER.
00026	MISSING TEMPERATURE SERIES KEY(S).

<b>Code</b>	<b>Error</b>
00027	NO CHANNEL SUPPLIED FOR TEMPERATURE SERIES KEY.
00028	TEMPERATURE SERIES KEY CHANNEL IS INVALID.
00029	TEMPERATURE SERIES KEY IS INVALID.
00030	NO CHANNEL SUPPLIED FOR TEST SERIES TEMPERATURE KEY.
00031	PROXY SERIES TEMPERATURE KEY CHANNEL IS INVALID.
00032	PROXY SERIES TEMPERATURE KEY IS INVALID.
00033	INVALID TEMPERATURE DEGREES DIFFERENCE.
00034	INVALID TEMPERATURE DEGREES TEST TYPE.
00035	DEGREES TEST MISSING INTERVAL THRESHOLD MUST BE A NUMBER FROM 0 TO 23.
00036	UNRECOGNIZED PARAMETER FOLLOWING TEST TYPE ON DEGREES COMMAND.
00037	INVALID DAY COMPARISON TYPE: MUST BE DOW, DT, OR ANY.
00038	MINUTES ALLOWED BETWEEN PEAKS MUST BE 0-1440.
00039	INVALID MAGNITUDE TEST WEIGHT: MUST BE BETWEEN 0 AND 1.
00040	INVALID SHAPE TEST WEIGHT: MUST BE BETWEEN 0 AND 1.
00041	UNKNOWN ENVIRONMENT FILE COMMAND.
00042	A SELECTION DATE RANGE IS REQUIRED.
00043	NO KEY WAS ENTERED FOR SUPPLIED LOAD SHAPE DAY.
00044	A TEMPERATURE DEGREES TEST WAS SPECIFIED, BUT NO TEMPERATURE CUT SERIES KEY WAS SUPPLIED.
00045	A TEMPERATURE CUT SERIES KEY WAS SUPPLIED, BUT NO TEMPERATURE DEGREES TEST WAS SPECIFIED.
00046	SUM OF WEIGHTS SPECIFIED FOR MAGNITUDE AND SHAPE TESTS NOT = 1.
00048	NUMBER OF HOLIDAYS IN SELECTION DATE RANGE EXCEEDS MAXIMUM (200).
00049	INVALID DATE ON LINE XX OF HOLIDAY FILE. (where XX is the line number of the holiday file)
00050	HOLIDAY FILE NOT SUPPLIED OR REQUESTED HOLIDAY FILE NOT FOUND.
00051	SEASON SCHEDULE NOT USABLE OVER SELECTION DATE RANGE.
00052	REQUESTED SEASON SCHEDULE NOT FOUND OR NOT USABLE FOR SUPPLIED LOAD SHAPE DAY.
00060	UNABLE TO OPEN INTERVAL DATABASE.
00061	PROBLEM GETTING SUPPLIED LOAD SHAPE DAY ENERGY.



<b>Code</b>	<b>Error</b>
00070	UNIT-OF-MEASURE CODE FOR PROXY DATA CUT SERIES INCOMPATIBLE WITH THAT FOR SUPPLIED LOAD SHAPE DAY.
00071	UNIT-OF-MEASURE CODE IS NOT CONSISTENT THROUGHOUT PROXY DATA CUT SERIES.
00080	NO QUALIFIED PROXY DAY FOUND IN PROXY DATA CUT SERIES.
00093	UNDEFINED FILE CONDITION IN PROXY DAY PROGRAM.
00101	DATA FOR SUPPLIED LOAD SHAPE DAY SERIES STOPS BEFORE REQUESTED DAY.
00102	DATA FOR SUPPLIED LOAD SHAPE DAY SERIES STARTS AFTER REQUESTED DAY.
00110	RECORDS FOR SUPPLIED LOAD SHAPE SERIES DO NOT SATISFY MERGE CRITERIA.
00111	LOAD DATA FOR SUPPLIED LOAD SHAPE DAY CONTAINS ALL MISSING INTERVALS.
00151	NO DATA FOUND FOR SUPPLIED LOAD SHAPE DAY SERIES.
00160	LOAD DATA NOT AVAILABLE FOR INTERVAL BEFORE SUPPLIED LOAD SHAPE DAY: SHAPE TEST CANNOT BE PERFORMED.
00162	UNIT-OF-MEASURE CODE FOR PROXY DATA CUT SERIES INCOMPATIBLE WITH THAT FOR SUPPLIED LOAD SHAPE DAY.
00164	UNIT-OF-MEASURE CODE IS NOT CONSISTENT THROUGHOUT PROXY DATA CUT SERIES.
<u>00165</u>	NEGATIVE UNIT-OF-MEASURE ENCOUNTERED IN TEMPERATURE CUT SERIES (ERROR CONDITION SHOULD BE FIXED, TEMPERATURE TEST BYPASSED)
<u>00201</u>	TEMPERATURE DATA NOT AVAILABLE FOR INTERVAL BEFORE SUPPLIED LOAD SHAPE DAY. TEMPERATURE ELIGIBILITY TEST WILL NOT BE PERFORMED.
<u>00202</u>	ALL TEMPERATURE DATA FOR DAY BEFORE SUPPLIED LOAD SHAPE DAY IS MISSING. TEMPERATURE ELIGIBILITY TEST WILL NOT BE PERFORMED.
<u>00210</u>	TEMPERATURE SERIES RECORDS FOR SUPPLIED LOAD SHAPE DAY DO NOT SATISFY MERGE CRITERIA.
<u>00215</u>	NON-TEMPERATURE UNIT-OF-MEASURE CODE IN SUPPLIED LOAD SHAPE DAY TEMPERATURE CUT.
<u>00251</u>	NO TEMPERATURE DATA FOUND FOR SUPPLIED LOAD SHAPE DAY.
<u>00260</u>	MISSING TEMPERATURE INTERVALS FOR SUPPLIED LOAD SHAPE DAY.
<u>00261</u>	INSUFFICIENT TEMPERATURE DATA FOR ONE OR MORE PROSPECTIVE PROXY DAYS.

---

<b>Code</b>	<b>Error</b>
<u>00262</u>	UNIT-OF-MEASURE CODE FOR PROXY TEMPERATURE CUT SERIES INCOMPATIBLE WITH THAT FOR SUPPLIED LOAD SHAPE DAY
<u>00263</u>	NON-TEMPERATURE UNIT-OF-MEASURE CODE IN PROXY TEMPERATURE CUT SERIES.
<u>00264</u>	UNIT-OF-MEASURE CODE IS NOT CONSISTENT THROUGHOUT PROXY TEMPERATURE CUT SERIES.
<u>00299</u>	INSUFFICIENT TEMPERATURE DATA AVAILABLE IN PROXY TEMPERATURE CUT SERIES.
<u>00300</u>	INSUFFICIENT TEMPERATURE DATA FROM DAY BEFORE SUPPLIED LOAD SHAPE DAY TO FILL MISSING FIRST INTERVAL(S).
00400	LOAD DATA NOT AVAILABLE FOR INTERVAL BEFORE DAY.
00904	INTERVAL DATA ERROR (specific message is based on the specific type of error).
99999	UNDETERMINED ERROR IN PROXY DAY PROGRAM.

---

---

## Proxy Day Tests and DST

Some of the tests performed by the Proxy Day DLL can be impacted based on Daylight Savings Time (DST). These include the Magnitude Test and the Shape Test.

### Magnitude Test

The magnitude test compares the candidate day's energy to the supplied day's energy on an hour by hour basis.

- If one of these days is short an hour (i.e. a 23 hour day) because it's a Spring DST change day, the hour in the other day that corresponds to the missing hour does not participate in the test.
- If one of these days has a repeated hour (i.e. a 25 hour day) because it's a Fall DST change day, the first of the repeated hours does not participate in the test.

### Shape Test

The shape test compares the candidate day to the supplied day by determining and comparing the energy differences between each consecutive pair of intervals on each day. The DST handling is similar to that for the Magnitude Test, but even if an interval is skipped due to DST, that interval still participates in this test by being subtracted from the interval that follows it.

For example, the supplied day is a 23 hour day and the interval from 2:00 to 3:00 is not present, and the candidate day is a 24 hour day whose energy values (starting at midnight on the Spring DST day) are a, b, c, d, e, etc. The Shape Test for the candidate day will use (b-a), (d-c), (e-d), etc. So while "c" has no corresponding interval in the supplied day and the (c-b) calculation is not part of this calculation, "c" still has some influence on the results of the test because the (d-c) calculation is used in the test.



# Chapter 8

---

## Configuring Oracle Utilities Load Profiling and Settlement Security

This chapter describes how to configure security to work with the Oracle Utilities Load Profiling and Settlement application, including:

- **Oracle Utilities Load Profiling and Settlement Security**

# Oracle Utilities Load Profiling and Settlement Security

This chapter describes the securable features of the Oracle Utilities Load Profiling and Settlement web application, including:

- **Profiling and Settlement Features**
- **Important Notes about Assigning Oracle Utilities Load Profiling and Settlement Permissions**

## Profiling and Settlement Features

These features enable the menu options on the Profiling and Settlement menu, and include:

- **Run Settlement:** Enables the Run Settlement menu option on the Profiling and Settlement menu.
  - **Schedule:** Enables the Schedule Settlement option on the Run Settlement screen.
- **Search Settlement:** Enables the Search Settlements menu option on the Profiling and Settlement menu.
- **Settlement Tab Set:** Enables the Settlement clip and screen on the Summary Account screen.
- **Run Reports:** Enables the Run Reports option on the Profiling and Settlement menu.
- **View Reports:** Enables the View Reports option on the Profiling and Settlement menu.
- **Actions:** Enables the Actions icon on the Settlement List and Aggregation screens.
  - **Run Aggregation:** Enables the Run Aggregation option on the Actions pop-up menu.
  - **View Aggregation:** Enables the View Aggregation option on the Actions pop-up menu.
  - **View Report:** Enables the View Report option on the Actions pop-up menu.
  - **Graph:** Enables the Graph/Download option on the Actions pop-up menu.
  - **Edit Schedule:** Enables the Edit Schedule option on the Actions pop-up menu.
  - **Delete:** Enables the Delete option on the Actions pop-up menu.

## Important Notes about Assigning Oracle Utilities Load Profiling and Settlement Permissions

By default, the Oracle Utilities Load Profiling and Settlement permissions restrict access to all Oracle Utilities Load Profiling and Settlement functions and screens. To allow users access to Oracle Utilities Load Profiling and Settlement functionality, create roles and enable appropriate permissions for each.

# Appendix A

---

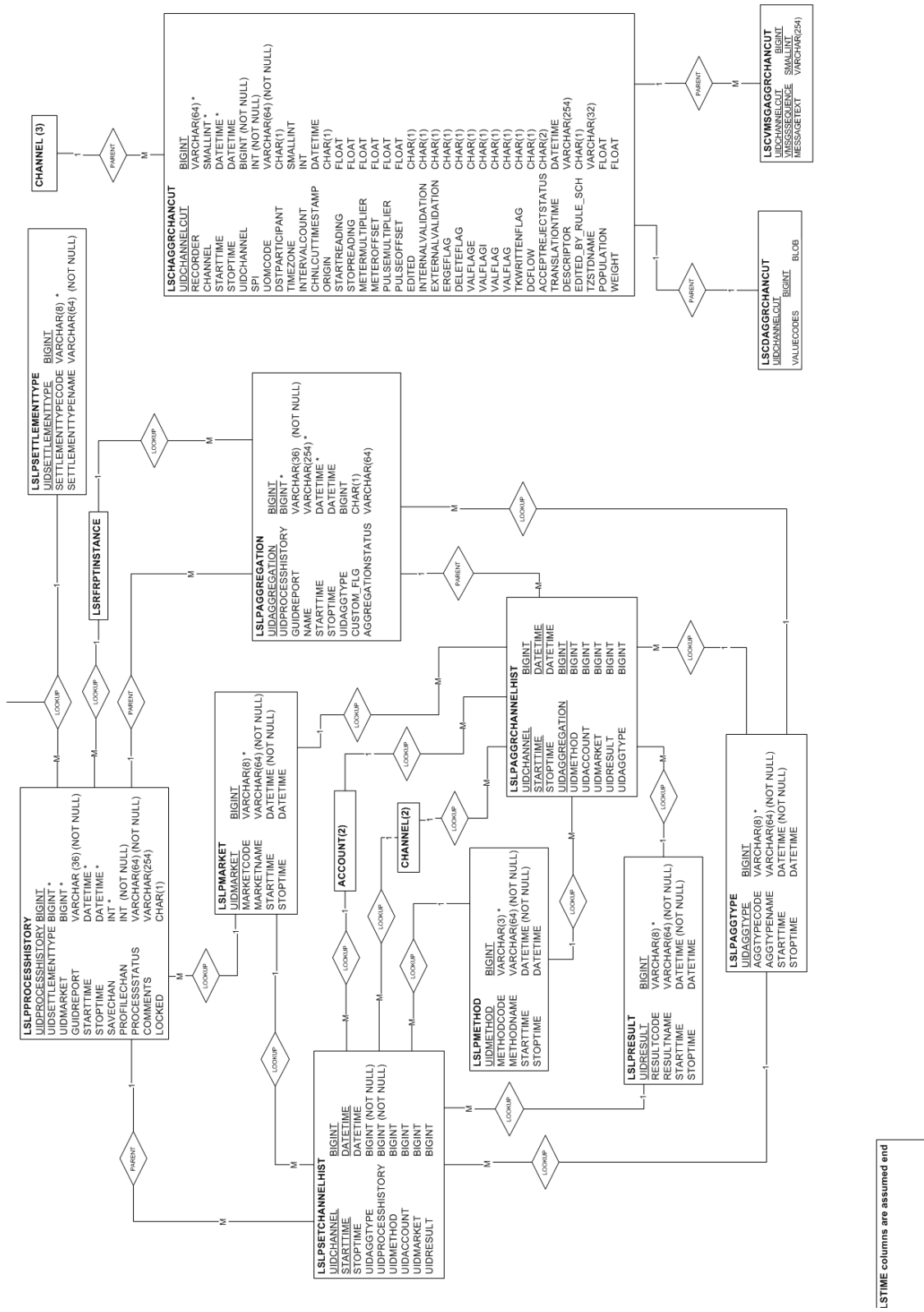
## Oracle Utilities Data Repository Load Profiling and Settlement Database Schema

This appendix includes a diagram of the Oracle Utilities Data Repository Load Profiling and Settlement database schema (v1.6.1.0.0) that provides details regarding the table and columns in the Oracle Utilities Load Profiling and Settlement schema, as well as the relationships between these tables in the Oracle Utilities Data Repository. This information is very useful when writing Rules Language statements or constructing database queries. This includes:

- **Oracle Utilities Load Profiling and Settlement Database Schema, page 1**
- **Oracle Utilities Load Profiling and Settlement Database Schema, page 2**

# Oracle Utilities Load Profiling and Settlement Database Schema, page 1

Utilities Load Profiling and Settlement v1.6.0.0.0  
 1 of 2  
 Utilities Proprietary & Confidential



LSLTIME columns are assumed end







---

---

# Index

## A

- account scalar factor 2-10
- Account Service History Table 2-6
- account-specific profile 2-4
- adjusted class profile 2-9
- adjusted static load profile 2-3, 2-9
  - to calculate class profiles 2-9
- adjusting a class profile 2-2
- aggregating individual records 2-10
- Aggregation Group Table 2-6
- aggregation groups
  - definition 2-5
  - used in the settlement process 2-6
- aggregation levels 2-6
  - user-defined 2-4
- aggregation types 2-6
- allocate UFE 2-10
- ancillary services
  - definition 2-8
- applying hourly loss factors 2-10
- applying loss factors 2-8
- applying losses 2-10
- applying profiles to accounts 2-2

## B

- bands 2-11
  - definition 2-7
- Bill History Table 2-11
- billing SCs 2-11
- BillingExpert 2-11

## C

- calculate class profiles for initial and final settlement 2-9
- calculating and allocating UFE 2-10
- calculating imbalances 2-7, 2-11
- calculating losses 2-4
- calculating UFE 2-5, 2-8
- CIS system 2-9
- class load profiles 2-9
  - developing 2-3
- class load shapes
  - dynamic modeling 2-3
- class profile
  - adjusted 2-9

- adjusting 2-2
  - scaling to account level 2-10
  - selecting 2-2
- class profile creation methods 2-3
- class profiles
  - calculation 2-9
  - creating 2-2
  - historical 2-2
  - multiplying by ratio 2-10
  - retrieving 2-10
- creating class profiles 2-2
- creating interval data for each market participant 2-8
- current system load 2-9
- customer class aggregation type 2-6

## D

- data aggregation 2-4, 2-10
- Database Installation 3-9
- Database schema diagram 3-11, A-1
- developing class load profiles 2-3
- differential allocation of UFE 2-5
- DisCo 2-5
- dynamic load profiling 2-3
- dynamic loss factor calculations 2-5
- dynamic modeling load profiling 2-3

## E

- entity-specific profile
  - generating 2-10
- Environment Files
  - Definition of 7-5
- ESP 2-5
- estimate of profiled load 2-10
- estimated customer load
  - calculating 2-10
- estimated monthly settlement 2-11
- estimates
  - fine-tuning 2-5
  - forcing to match system load 2-5
- estimating
  - missing data 2-10
- estimations
  - reconciling to system load 2-2

## F

- FERC 2-6
- final settlement 2-11
- final settlement stage 2-4
- financial settlement 2-11
  - calculations 2-8
  - factors 2-8
  - process 2-8, 2-11
- fine-tuning
  - estimates 2-5

## G

- generating an entity-specific profile 2-10
- generation losses 2-5

## H

- historic class load profile 2-9
- historical class profiles
  - definition 2-2
- historical interval data 2-2
- historical system load 2-9
- Holiday File
  - Setting up 7-9
- hourly load 2-10
- hourly loss factors
  - applying 2-10
- hourly ratio 2-10

## I

- imbalance
  - definition 2-8
- imbalance calculations 2-11
- imbalance curve 2-7, 2-11
- imbalance outside of the band 2-11
- imbalance shapes 2-7
- imbalances
  - calculating 2-7
- Installing the LPSS Software 4-2
- interval data recorder accounts 2-10
- interval data recorder customers 2-10
- interval data validation 2-9
- ISO
  - definition 2-6

## L

- LDC 2-5
- load profiling
  - definition 2-2
  - dynamic 2-3
  - Proxy day 2-3
- load research 2-2
- load research data 2-2
- load shape
  - profiled 2-10
- loss calculations 2-4
- loss class aggregation type 2-7
- loss factors 2-5
  - applying 2-10
  - hourly 2-10

LPSS Overview 1-1

## M

- market participant load 2-4
- market participants that serve as aggregation groups 2-5
- matrix penalty amounts 2-11
- meter cycle 2-9
- meter read accounts
  - profiling 2-10
- meter reads
  - pre-aggregation 2-9
- missing data
  - estimating 2-10
- modeling class load shapes 2-3
- monthly settlement 2-1
  - estimated 2-11

## N

- net imbalance 2-11
- net imbalance shape 2-11
- net UFE 2-11

## O

- Oracle Utilities Data Repository 1-2
- outside the band 2-11
- Outside-Band Charge 2-11
- overview
  - load profiling 2-2

## P

- penalty for faulty forecasting 2-7
- penalty matrix 2-11
- penalty percentages 2-11
- performing financial settlement 2-11
- performing monthly settlement with LPSS 2-9
- pre-aggregate scalar data 2-4
- pre-aggregation of meter reads 2-9
- pricing matrix 2-11
- processing groups 2-4
- profile
  - static 2-9
- profile application
  - scalar energy reading 2-4
  - unadjusted 2-4
  - usage factor 2-4
- profile application methods 2-4
- profile application process 2-4
- profile available aggregated meter read records 2-10
- profile creation 2-3
  - definition 2-2
- profile creation methods 2-3
- profile creation process 2-2
- profile status flag 2-10
- profiled based on usage factor 2-10
- profiled load
  - estimating 2-10
- profiled load shape 2-10
- profiles
  - aggregating 2-2
  - applying to accounts 2-2

- profiling 2-2
  - purpose 2-2
- profiling meter read accounts 2-10
- profiling methods 2-2
- profiling routines 2-2
- Proxy Day load profiling 2-3
- Proxy Day selection 2-3
- purpose of creating interval data for each market participant 2-8

## R

- rate class aggregation type 2-7
- ratio 2-9
  - current system load as a percentage of historical system load 2-10
  - each SC's hourly load to the total of all SCs' hourly loads 2-10
  - energy in the account meter read divided by energy in the class profile 2-4
- ratio of a particular estimate's load in a specific hour to total of all estimated loads in that hour 2-5
- ratio of monthly meter energy divided by total energy in class profile 2-10
- reconcile estimations to system load 2-2
- referential integrity checks during the Import process 2-9
- REP 2-5
- reporting 2-7, 2-11
- Rules Language 2-4

## S

- sample
  - statistical 2-2
- sampling 2-2
- SC imbalance 2-11
- SC schedules 2-9
- scalar data
  - pre-aggregation 2-4
- scalar energy reading profile application 2-4
- scaling class profile to the account level 2-10
- Schedule Coordinators 2-5
- Season File 7-7
- selecting a class profile 2-2
- Service Coordinators 2-11
- settlement
  - financial 2-11
- settlement schedules 2-2
- SIC 2-9, 2-11
- static load profile 2-3
  - adjusted 2-3, 2-9
- static profile 2-9
- Statistical Analysis Extension of Oracle Utilities Load Analysis 2-3
- System Incremental Cost (SIC) 2-9, 2-11
- system load
  - current 2-9
  - historical 2-9

## T

- tables
  - Account Service History 2-6
  - Aggregation Group 2-6
  - Bill History 2-11
  - Usage Summary 2-10
- Time-of-Use periods 2-7

- total system UFE
  - allocating 2-8
- transmission and distribution charges 2-8

## U

- UDC 2-5
- UFE 2-10, 2-11
  - calculating 2-4
  - calculating and allocating 2-10
- UFE allocation 2-5
- unadjusted profile application 2-4
- update LPSS Customer data 2-9
- update LPSS Interval data 2-9
- usage factor
  - profiling based on 2-10
- usage factor profile application 2-4
- usage factor profiling 2-2
- usage profiles 2-2
- Usage Summary Table 2-10

## V

- validate imported data 2-9
- Validation, Editing, and Estimation (VEE) 2-9
- VEE 2-9
- voltage class aggregation type 2-6

## X

- X410 - Load Data Reporter
  - Input files
    - Holiday File 7-9
    - Season File 7-7

