Oracle Utilities Load Analysis

User Guide

Release 1.11.1.1 for Windows

E18234-09

June 2017

ORACLE

Oracle Utilities Load Analysis User's Guide, Release 1.11.1.1 for Windows

E18234-09

Contents

Preface

Chapter One

Chapter Two

Chapter Three

Preface

Welcome to Oracle Utilities Load Analysis

This preface is designed to help you understand the contents and purpose of this manual, as well as the other resources available to you.

Welcome

Welcome to Client/Server Oracle Utilities Load Analysis (also called Oracle Utilities Load Analysis). This set of software programs has been designed by Oracle Corporation to help utilities collect, manage, and analyze load research data. Evolved from the original mainframe version of Oracle Utilities Load Analysis, it adds the advantages of personal computers and the latest developments in network architecture, as well as many new tools and features. Its design reflects more than 20 years of experience in software development for electric and gas utilities.

What Does this Guide Cover?

This guide explains how to use the graphical user interface of Oracle Utilities Load Analysis to "submit jobs" – that is, how to select a desired program, create and specify the necessary input files, run the job, and view the results. While this guide covers the mechanics of ***how*** to use Oracle Utilities Load Analysis, it does not explain ***what*** you can accomplish with the system. It does not describe the features, functions, or operation of individual programs. (For example, it does not cover the parameters that you must specify within specific input files for each program.) For that information, you must refer to the *Oracle Utilities Load Analysis Introductory Guides, Volumes I and II*. Use this guide as a secondary companion piece to those documents. In fact, if you are a new Oracle Utilities Load Analysis user, you'll probably want to have both manuals open on your desk as you begin to explore the system.

How is this Guide Organized?

This guide consists of eight chapters and five appendices:

Chapter One - Overview of Oracle Utilities Load Analysis

Contains information of special interest to both new and existing Oracle Utilities Load Analysis users. For new users, it defines some basic Oracle Utilities Load Analysis terms and concepts.

Chapter Two - Getting Started

Explains how to open the Oracle Utilities Load Analysis interface, and then provides a brief overview of the function of each element you'll see on the "desktop" (menu commands, tool buttons, etc.). It also describes security restrictions, and how you can customize the system to your needs and environment using the setup options.

Chapter Three - Working with Jobs

Takes you step by step through the process of submitting a job and viewing the results - the primary tasks you'll perform with Oracle Utilities Load Analysis.

Chapter Four - Working with Input Files

Explains how to open, edit, and create Environment, Control, and Common files.

Chapter Five - Managing Data

Explains how to put interval data in a database on the network server, so that it is available for analysis using the Oracle Utilities Load Analysis programs. It also describes how you can use some new tools for moving, copying, and deleting cuts (statistics and interval data records).

Chapter Six - The Program Sequencer

Explains how to use the Program Sequencer to "chain" together a series of Oracle Corporation, user-written, and/or third party programs in a customized sequence. You can run the sequence immediately using the Oracle Utilities Load Analysis Graphical User Interface (GUI), or you can schedule it for delayed future execution and/or regular periodic execution using the Windows Schedule Service.

Chapter Seven - The Repeater

Explains how to use the Repeater to automatically run an analysis program multiple times, each time with a different Control File and/or Environment File. This is useful if you want to apply a variety of analysis parameters to different sets of cuts without having to resubmit the job for each variation.

Appendix A - Programs Available via Oracle Utilities Load Analysis

Lists each of the Oracle Utilities Load Analysis programs you can use via the Graphical User Interface (GUI), along with a brief description of the program's application and a cross reference to the appropriate user manual for specific instructions.

Appendix B - Oracle Utilities Load Analysis Naming Conventions

Lists each of the Oracle Utilities Load Analysis program/file naming conventions that you will see in this manual and on the screen, including optional extensions.

Appendix C - Oracle Utilities-Supplied Input Files

Lists each of the test input files that Oracle Corporation supplies with Oracle Utilities Load Analysis. You can use these files as a starting point for your own work.

Appendix D - Oracle Utilities Load Analysis Sequencer Program Input Tables and Keywords

Summarizes the commands used to create "SEQ" files for Oracle Utilities Load Analysis programs. This information is intended to be used in conjunction with **Chapter Six**: **The Program Sequencer**.

Conventions Used in this Manual

This guide assumes that you have a working knowledge of your computer and network, as well as their operating conventions. For information about Windows operating conventions, see your Windows documentation.

The following conventions are used in this guide.

Mouse Buttons:

All instructions refer to the left mouse button unless otherwise indicated. Use the *right* mouse button only when specifically instructed.

Typographic Conventions:

Information that you type is shown in a typeface called Courier New, which looks like typewriter type. For example: `Oracle Utilities`.

The names of keys you press are represented by small capital letters. For example: ENTER. Occasionally, in this manual and on the Oracle Utilities Load Analysis interface, you will see two key names with a plus sign (+) between them. This indicates that you should press and hold down the keys in the order specified and release them together. For example, ALT+F5 means that you should press the ALT key, and while holding it down, press F5.

A sequence of menu commands you select is represented by the name of each menu or command followed by a small arrow, in **boldface** type. For example, **File->Print** means select **File** from the desktop or window menu bar, then **Print** from the pull-down menu that appears.

How To Get Help

Customer Support

You can contact Oracle Support personnel at http://metalink.oracle.com. My Oracle Support offers you secure, real-time access to Oracle experts on the complete Oracle Utilities Load Analysis system. It also provides ground breaking personalized & proactive support capabilities that help reduce unplanned down time and improve system stability. Leverage the Internet for immediate access to 24/7 support and get the critical and timely information you need for running your business.

Additional Documentation

In addition to the *Oracle Utilities Load Analysis Introductory Guides,* another reference you may find helpful is the *Oracle Utilities Load Analysis Quick Reference Guide* — a concise summary of program names, input file commands and parameters, standard codes, and other important information for the basic Oracle Utilities Load Analysis system and its extensions.

User Feedback

This document will be refined and updated over time. We would appreciate your help in improving it. If you have suggestions for adding new topics or for clarifying existing explanations, please let us know by calling the Help Line or by completing and returning the Customer Feedback form at the back of this guide.

Chapter One

Overview of Oracle Utilities Load Analysis

This chapter provides a brief overview of Oracle Utilities Load Analysis. It begins by generally describing how you work with the system — an especially useful section for new users since it covers some key terms and concepts referred to throughout the rest of this guide. Next you will find a brief overview of the features introduced with each new release.

How Do You Work with Client/Server Oracle Utilities Load Analysis?

The Oracle Utilities Load Analysis system is a suite of software programs developed by Oracle Corporation to help utilities conduct successful load research programs. It incorporates programs for interval data input, editing, validation, analysis, reporting, graphing, and more. (A complete list of the programs available with this version of Oracle Utilities Load Analysis is provided in **Appendix A**: **Programs Available via Oracle Utilities Load Analysis**.)

You accomplish tasks by selecting a program to process the data, specifying **input files**, identifying the **database** that holds the data with which you wish to work, and viewing the **outputs** (also called **Results**). Users interact with the system via a point-and-click graphical user interface (GUI).

Input Files

Most Oracle Utilities Load Analysis programs require you to specify two types of input files: Control Files and Environment files. A **Control File** is often a list of the specific data records (called "cuts") that you want to apply the program to, such as customer interval data or computed statistics. An **Environment File** is the set of commands and parameters that define how the program will process the data.

Many programs, such as the analysis and reporting programs, also use one or more **Common files** to define various schedules:

- ***Holiday File***: a list of all national and local holidays observed in your service territory.
- ***Time-of-Use Schedule File***: a schedule of on-, off-, and shoulder peak periods.
- ***Season File***: a schedule applying different TOU schedules to seasons throughout the year.
- ***Demand Period File***: a schedule of "on-demand" periods.
- ***User-Specified Day File***: a schedule of user-defined days to be averaged together in an analysis.
- ***Billing Cycle File***: a schedule of billing cycles for the Billed Energy Program.
- ***Peak Days File***: a schedule of Peak Days for the Daytype Analysis Program.

These are called "Common files" because they reflect the policies of your facility, are typically accessed by all Oracle Utilities Load Analysis users, and are located in your Common\Data folder (more on that later).

Databases

Oracle Utilities Load Analysis has several types of interval databases to keep track of data in the different stages of the input-validation-analysis cycle. The **CLDB** (Current Load Database) contains raw customer interval data that is in the process of being verified. The **ALDB** (Archive Load Database) typically contains load data that has been checked and edited for completeness and reliability, and is considered ready for analysis. The **ELDB** (Extracted Load Database) is the working database for the analysis programs. You can use the **SLDB** (Statistics Load Database) to archive selected subsets of the ELDB. The **GLDB** (Generator Level Database) is the working database for the Cost of Service Interface. The **RLDB** (Rates Level Database) is the archive version of the GLDB.

Your facility may have multiple databases of each type.

Output Files

Most Oracle Utilities Load Analysis programs produce three types of output files:

- ***Reports***: Oracle Utilities Load Analysis combines all reports into one file called "SYSPRINT". The system adds an index to the end of the file, so you can quickly locate a desired report.
- ***Data***: Data files can include individual customer statistics files and keylists.
- ***Log***: System log containing information about the program run. This is useful if there is a problem with the job.

Chapter Two

Getting Started

This chapter explains how to open and close the Oracle Utilities Load Analysis desktop, your primary work area. To help you get oriented quickly, it also gives a brief overview of each command and tool available to you on the desktop, as well as the use of some function keys. Finally, it describes how you can customize your system, and how you can view a list of your system privileges.

This chapter includes:

- **Starting and Ending a Oracle Utilities Load Analysis Session**
- **Getting Acquainted with the Oracle Utilities Load Analysis Desktop**
- **Sunrise - Sunset Calculator**
- **Defining Holidays, Time Of Use Periods, Seasons, and User Days in the Oracle Utilities Data Repository**
- **Restrictions**

Starting and Ending a Oracle Utilities Load Analysis Session

This section explains how to start and close the Oracle Utilities Load Analysis Graphical User Interface and log on to the server.

How to Start Oracle Utilities Load Analysis:

1. At the Windows desktop, click the **Start** button. Then select **Programs->Oracle Utilities->Load Analysis Client 01.11.0.00.00**

   (If the options described above do not appear on your system, refer to your Windows documentation, or consult your System Administrator. There are many ways to start a program in Windows, and your system may be set up with different programs groups or other customizations.)

   After a few moments, the Oracle Utilities Load Analysis desktop appears on your screen. All Oracle Utilities Load Analysis functions can be accessed from this window.

2. *Optional.* Log on to the server. (This is the default state — logged on.)

   > **Note**: The server that you log on to, and whether or not logon is automatic, is determined by your setup options. See **Options** on page 2-6 for details.

   Almost all work that you'll perform with Oracle Utilities Load Analysis requires that you log on to the Oracle Utilities Load Analysis Server.

   Depending upon how your system is configured, the system may automatically log you on to the server when you open the Oracle Utilities Load Analysis interface. You can tell whether or not you are logged on by looking at the Logon/Logoff button in the toolbar.

   Logged Off:  Logged On:  (pushed in)

   When you have successfully connected to the server, a message appears in the information box at the bottom of the desktop window, "`User (your network ID) logged on CSL* server (server name)`". You are now ready to begin working with Oracle Utilities Load Analysis.

How to Close the Oracle Utilities Load Analysis Interface:

1. Log off the server by clicking the **Logoff** button.
2. Close the interface. There are four alternate methods for closing the Oracle Utilities Load Analysis interface (all produce the same result):
   - Select **File->Exit**.
   - Press ALT + F4.
   - Press CTRL + F12.
   - Click the Close button (**X**) in the upper right corner of the window.

Getting Acquainted with the Oracle Utilities Load Analysis Desktop

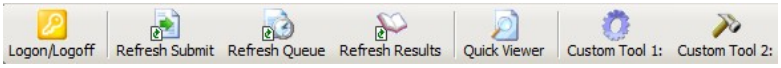
The Oracle Utilities Load Analysis desktop is your starting point for all Oracle Utilities Load Analysis functions. You can interact with the system either via the Menu Bar commands or the Toolbar.

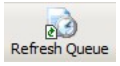
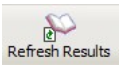
> **Note**: In addition, some functions can be performed using the Function Keys on your keyboard. See **Function Keys** on page 2-5 for more information.

The Toolbar gives you quick access to Oracle Utilities Load Analysis's most frequently used tools and commands. Just click the desired button to submit a job, view results, and more.

Toolbar

The following describes the function of each Toolbar button. Additional details are provided in the remainder of this guide.



Icon	Description
	**Logon/Logoff**: Log on to or log off from the Oracle Utilities Load Analysis server. Also acts as a status button; that is, if the button is bright, you are already logged on, if it is dark, you are logged off.
	**Refresh Submit**: Refreshes the submit screens. See **Chapter Three**: **Working with Jobs**.
	**Refresh Queue**: Refreshes the queue of previously-submitted jobs on the server. See **Chapter Three**: **Working with Jobs**.
	**Refresh Results**: Refreshes files in Oracle Utilities Load Analysis job directories; that is, the results of completed Oracle Utilities Load Analysis jobs. Also gives you access to both Display Tool and other viewers. See **Chapter Three**: **Working with Jobs**.
	**Quick Viewer**: Provides brief explanations of selected program functions.
	**Data Browser**: Opens the Data Browser. See **Using the Data Browser** on page 2-13.
	**Custom Tool 1:** Open a configured Windows accessory. See **Options** on page 2-6.
	**Custom Tool 2:** Open a configured Windows accessory. See **Options** on page 2-6.

Menu Commands

Any of the commands and tools that you can activate using the Toolbar (just described) can also be accessed using Menu bar. In addition, there are a few commands that you can use only via the Menu bar:

Command	Description
File-› Transfer Type Utility	Opens the Oracle Utilities Load Analysis Type Transfer Utility, used for convert multi-typed information from one CSLSINFO.MDB to another. See **Chapter Four*: Working with Input Files***.
File-› Setup	Configures Oracle Utilities Load Analysis to your specific needs and environment.
File-› Server Configuration	Available only to System Administrator. See the *Oracle Utilities Load Analysis Configuration Guide*.
File-› Exit	Close the Oracle Utilities Load Analysis interface.
Tools-›Logon	Logs the current user on to the server.
Tools-›Logoff	Logs the current user off the server.
Tools-›Restrictions	View a list summarizing which programs and databases you have been granted access to by the System Administrator. See **Restrictions** on page 2-20.
Tools-›Tool1	Opens the tool specified as Custom Tool 1 on the Tools tab on the Options dialog. See **Options** on page 2-6.
Tools-›Tool2	Opens the tool specified as Custom Tool 2 on the Tools tab on the Options dialog. See **Options** on page 2-6.
Tools-›Interval Data Manager	Opens Internet Explorer to the URL specified as the "IDM Home Address" on the Options tab of the Setup dialog. See **Options** on page 2-6.
Tools-›SunRise-Set Calculator	Opens the Sunrise - Sunset Calculator. See **Sunrise - Sunset Calculator** on page 2-10.
Tools-›Edit INI File	Opens the .ini file for editing.
Tools-›Options	User-selectable options for configuring Oracle Utilities Load Analysis to your needs and environment See **Options** on page 2-6.
Additional Tools	These tools can be chosen in the Options menu, as explained above.
Help	Displays a drop-down list of options, including **User Documentation** (a drop-down list of Oracle Utilities Load Analysis manuals that can be accessed in PDF format), **About...** *(*About Oracle Utilities Load Analysis*),* and a link to the Oracle Utilities Support web site.

Function Keys

Some functions can be performed in Oracle Utilities Load Analysis by use of the Function Keys. Some of these are unique to Oracle Utilities Load Analysis, and some are basic Windows functions used in Oracle Utilities Load Analysis.

Function Key	Result
Ctrl + F12	Exits the Oracle Utilities Load Analysis application
Alt + F4 (Windows function)	Exits the Oracle Utilities Load Analysis application
F5	Opens Internet Explorer to the URL specified as the "IDM Home Address" on the Tools tab of the Setup dialog. See **Tools** on page 2-6.
F6: For input files:	Allows you to select and include a Control File or Environment File in the File list box. Use a file not in your local data folder structure. This file is for one-time use only; the selection will be reset when you change the Submit panels
F6: For output files:	Allows you to specify path name together with output filename
F7	Opens the Server Configuration file

Options

Oracle Utilities Load Analysis makes it possible to configure the system to your particular needs and environment, using **Tools›Options**. When you select **Options**, the system displays a set of tab pages, as described below. The default values are shown here. You can change these settings as desired, because they affect work on your workstation only.

File Paths

This tab defines the path to the directory containing your input files.

- **Ctl and Env Files**: The path and directory containing your Control and Environment files. Oracle Utilities Load Analysis uses this path when displaying the names of available files in the **Submit Form** and **File Composer** windows.
- **Results Project Path:** The path and directory containing your project folders. Oracle Utilities Load Analysis uses this path when displaying the names of available folders and files in the Results panel.
- **Sequencer Files Path**: The path to the directory containing your sequencer files. Oracle Utilities Load Analysis uses this path to populate the Sequencer Files drop down in the "SEQR Sequencer" program panel
- **Sequencer Input Files Path**: The path to the directory containing your sequencer input files. See **Copying Files to the Sequencer Input Folder** on page 6-4 for details about enabling this option.

Checking the **Maintain Window State** box causes each window's state (as it pertains to minimization/maximization) to be maintained; that is, the windows will not maximize by themselves if this item is checked. If left unchecked, default Windows behavior remains in effect.

Tools

This tab defines the **Custom Tools** accessories found in the Oracle Utilities Load Analysis toolbar. By default, the entries here point to Windows accessories that are supplied with your operating system, but you can opt to use other programs if you have them.

Database Profiles

To make database connections easier for users who access multiple databases, the application saves the previous five database connections and displays them in a **Profiles** drop down list on the initial login screen and on the General tab of the File Setup window. In order to enable this functionality you must include the following setting in your .INI file:

```
enableDBProfiles=1
```

Once the setting has been changed, restart the client GUI. The initial database login screen will now contain a drop-down for profiles.

A new profile is created each time you provide a new, unique combination of database user, data source, or qualifier. You can also save your password for each profile by selecting the **Save Password** option on the login screen or on the General tab of the File Setup window. Selecting a profile in the drop down list updates the current database connect information to the selected profile. To delete a profile, select the profile in the drop down list and press the Delete key.

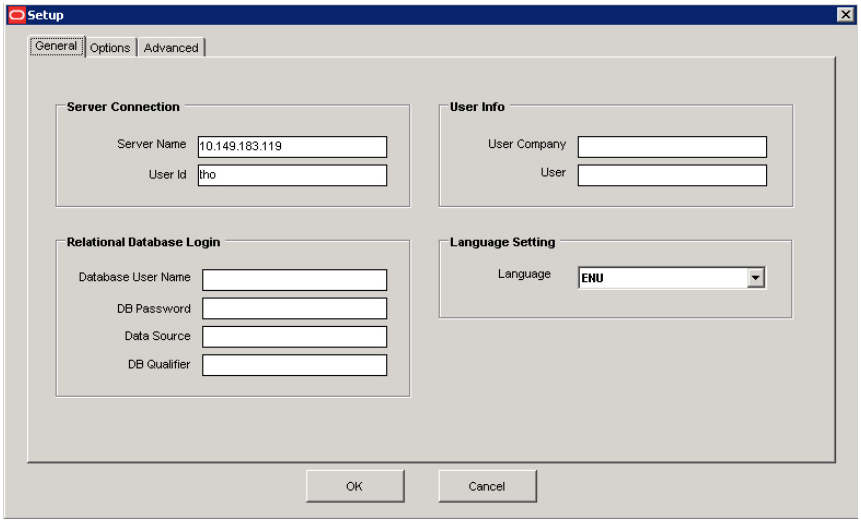
Setup

When you select **File->Setup**, the system displays another set of tab pages for customizing your configuration. Because most of the settings on these tab pages involve resources used by all Oracle Utilities Load Analysis users, they are typically set up for you by the System Administrator when Oracle Utilities Load Analysis is installed. (The values displayed for your system may differ from those shown here.)

The Setup window is organized into three tabs:

- **General** – This tabs contains general user as well as connection/database login information.
- **Options** – All available Client GUI configuration options are stored in this tab.
- **Advanced** – Stores advanced configuration settings that normally do not require changes.

General

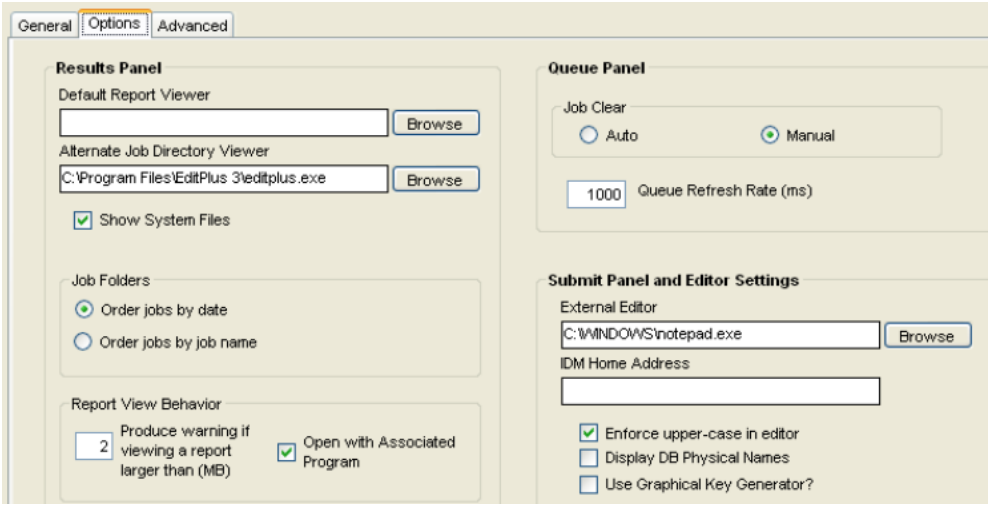


- **Server Name**: The network name of the Oracle Utilities Load Analysis server.
- **User Id**: The ID of the user that will be using this workstation. The ID here should match an entry in the CSLSTAR.USR File. The access privileges set in that file will determine what functions the user identified here can have from this workstation.
- **User Company** and **User Name:** The information in these fields appears on the Oracle Utilities Load Analysis title screen when the application is launched. It has no other purpose.
- **Profiles:** This field is displayed when you have turned on database profiles in the .INI file. The Profiles drop down list displays the last five databases that you have connected to. Selecting a profile updates the current database connection to the profile. See **Database Profiles** on page 2-6 for more information.
- **Database User Name**: User ID used to connect to the Oracle Utilities Data Repository.
- **DB Password**: Database password used to connect to the Oracle Utilities Data Repository. Note: This password must be supplied at login if any other RDB connection fields have been populated.
- **Data Source**: The Data Source name for the Oracle Utilities Data Repository.

  For Oracle databases, this is the Oracle TNS Name for the data source, from the TNS_NAMES.ora file (typically located in the \\<machine>\oracle\network\admin directory)
- **DB Qualifier**: Database qualifier used to connect to the Oracle Utilities Data Repository.

- **Save Password:** This field is displayed when you have turned on database profiles in the .INI file. Select this option if you want your password to be saved as part of your database profile. See **Database Profiles** on page 2-6 for more information.
- **Language:** The language setting for the application.

Options

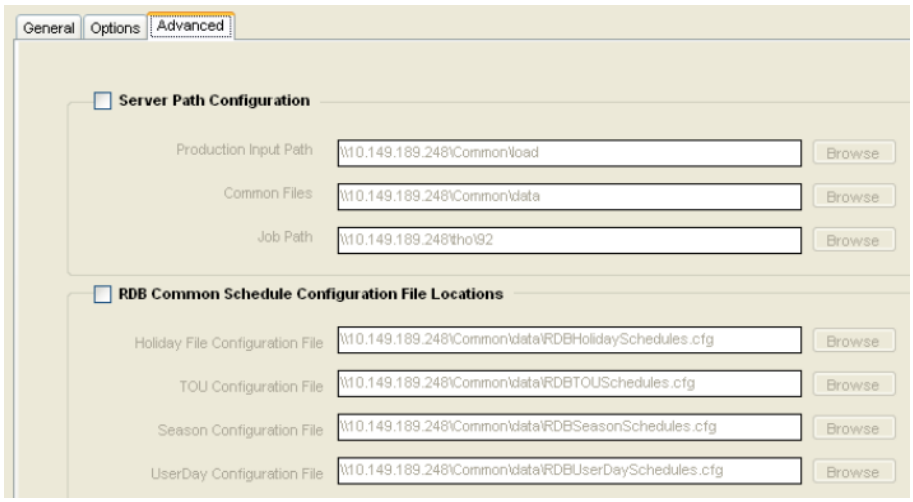


- **Default Report Viewer**: Oracle Utilities Load Analysis allows you to use a third-party program (such as Microsoft Word or Notepad) to view REPORT.HTML files. If you want to use this feature, specify the full path and executable name of the desired program, and uncheck "Open with Associated Program."
- **Alternate Job Directory Viewer**: The full path and executable name of an alternate program used to view results files in the job directory.
- **Job Folders:** Select how to sort results, by Create Date or Job Name (default). You may change this setting by right-clicking a Job Folder in the Results panel and selecting "Order by Date."
- **Report View Behavior**: These options allow you to specify how you view reports:
  - **Produce warning if viewing a report larger than (MB)**: Warns users if they are about to view a report that is very large. The default is 10 MB and a popup a message will appear allowing the user to cancel the load if the file is larger than the preset limit.
  - **Open with Associated Program**: Check this option to open Report files with the program associated with the file type. (Recommended.)
- **Queue Panel**:
  - **Job Clear**: Select how to clear jobs, Auto (automatically clear jobs) or Manual (manually clear jobs).
  - **Queue Refresh Rate**: Specify how often the queue is refreshed from the server (in milliseconds).
- **External Editor**: The full path and executable name of the program (i.e., Notepad) used to edit files such as INP or LSE files.
- **IDM Home Address**: The URL of the Oracle Utilities Energy Information Platform that includes Interval Data Manager (i.e., http://servername/lodestar/ccs). To open the Oracle Utilities Energy Information Platform at this URL, select **Tools->Interval Data Manager** (or click **F5**).

- **Enforce upper-case in Editor**: When this option is selected, any and all edits are done in upper-case letters. Editing in lower-case or mixed-cased is disabled. (Recommended.)
- **Display DB Physical Names:** To display the physical names of databases (set up in the CSLSTAR.DB file), click **Yes** (database Descriptions appear in brackets "<< >>"). To display database Descriptions only, click **No** (default).
- **Use Graphical Key Generator?:** Select whether or not to use the Graphical Key Generator. (Not recommended.)

Advanced

These fields are by default disabled and are automatically configured by the client, based on the user's current server and User ID information. Unless you have special configuration settings, you should not need to change settings here. Doing so can adversely affect your application's job submissions to the server. Values can be manually overridden by clicking on the associating check box and then changing the values.



- **Server Path Configuration**: In this screen, you can either enter the paths in the fields or click Browse... to search the network.
  - **Production Input Path:** Path to the directory on the server that holds the Production Input files.
  - **Common Files:** Path to the directory on the server that holds the Common files.
  - **Job Path:** Path to the current user's Job folder.
- **RDB Common Schedule Configuration File Locations**: These options specify configuration files used by Oracle Utilities Load Analysis, when accessing data in the Oracle Utilities Data Repository.
  - **Holiday File Configuration File:** A configuration file that specifies holiday lists stored in the Oracle Utilities Data Repository.
  - **TOU Configuration File:** A configuration file that specifies that Oracle Utilities Load Analysis should use Time-of-Use schedules defined in the Oracle Utilities Data Repository.
  - **Season Configuration File:** A configuration file that specifies that Oracle Utilities Load Analysis should use Time-of-Use schedules defined in the Oracle Utilities Data Repository.
  - **UserDay Configuration File:** A configuration file that specifies user day lists stored in the Oracle Utilities Data Repository.

Sunrise - Sunset Calculator

The Sunrise - Sunset Calculator is a tool that allows users to calculate sunrise and sunset times for any point on the globe. This can aid generating profiles that correspond to daylight or darkness hours.

How to calculate sunrise and sunset for a specific location over a specified data range:

1. Select **Tools->SunRise-Set Calculator**.
2. Select the location in which you wish to calculate sunrise-sunset times for. If your location isn't already defined, you add your location by doing the following:
   a. Enter a name for your new location in the **Select Location** field.
   b. Enter the latitude for the location in the **Latitude** field. Latitudes should be entered as <degrees>.<minutes> North. For example, a latitude of 54 degrees, 19 minutes North would be entered as "054.19." Coordinates South are designated with negative values. For example a latitude of 54 degrees, 19 minutes South would be entered as "-54.19".
   c. Enter the longitude for the location in the **Longitude** field. Longitudes should be entered as <degrees>.<minutes> West. For example, a latitude of 87 degrees, 54 minutes West would be entered as "87.54." Coordinates East are designated with negative values. For example a latitude of 87 degrees, 54 minutes East would be entered as "-87.54".
   d. Enter the Time Zone Standard Name for the location in the **TZ Standard Name** field. U.S. values are EST, CST, MST and PST. If you require a timezone other than the default, you will need to configure your LSCALENDAR.CFG.XML file located in your /CFG directory.
   e. Click "Save Location" once you've provided all necessary input for your new location.
3. *Optional*. Enter any necessary adjustments (in numbers of minutes) for the location in the Sunrise and Sunset fields in the **Adjustments** box. These allow for minor adjustments to the sunrise and sunset times to account for obstructions or other factors that may impact when the sun rises or sets in the location.
4. Select the **Start Date** and **Stop Date** for the date range.
5. Click **Calculate**. The sunrise and sunset times for the supplied date range appear in the Sunrise Sunset Results dialog.

Sunrise-Sunset Results

Sunrise - Sunset times are calculated for each day in the date range provided. You should do a quick check that these times are accurate, that sunrise - sunset times are occurring the expected approximate times. If not, you should be go back and check your location inputs and make adjustments as necessary.

With these Sunrise - Sunset results, you can:

- Export results to a COSI File: Results are exported as a COSI file that can be used with the SUN command in the G130 Control File used by the Cost of Service Interface - Electric (COSI) program.
- Export results to CSV: Results are exported into a comma-separated (CSV) file
- Create a Profile: Create a cut profile and import it into a database or save it in LSE format.

Example: COSI format

```
SUN
07:08 17:47
07:09 17:45
07:10 17:44
07:12 17:42
```

Example: CSV format

```
Sunrise,Sunset
10/24/2004 7:08:31 AM,10/24/2004 5:47:01 PM
10/25/2004 7:09:45 AM,10/25/2004 5:45:37 PM
10/26/2004 7:10:59 AM,10/26/2004 5:44:15 PM
10/27/2004 7:12:14 AM,10/27/2004 5:42:54 PM
```

To export results, click either **Export COSI File** or **Export CSV**. A Save As dialog opens allowing you to save the file.

Create a Sun Profile

Sun profiles are interval data cuts that represent usage based on the sunrise-sunset calculations.

How to create a Sun profile:

1. Click **Create Profile**.

   The **Sun Profile Generator** window opens.

2. Specify the **Profile Destination** for the profile:
   - Select **Database** if you want to write the profile to a database. Select the database you want to import the file into in the **Database** drop-down list.
   - Select **LSE** if you want to create an LSE file. Specify the name and location in which you wish to create your LSE file in the **Output LSE Filename** field.
3. Supply **Function Parameters** specific to the profile you wish to create:
   - **RecordedID**: Enter the Recorder ID, name, or CUSTID for this profile.
   - **Channel**: Specify the channel for this profile. The default is 1.
   - **SPI**: Specify the aggregation level in which you wish to create this profile. The default is 900 Seconds Per Interval.
   - **Energy to allocate (KWh)**: Specify the desired total energy to allocate for this profile.
   - **Descriptor**: Specify the description attached to this profile. The description can contain up to 80 characters.

- **Unit of Measure**: Select the unit of measure to assign to this profile. The default is KWh.
- **Ramp Duration (Seconds)**: Street Lighting doesn't consume at 100% energy usage instantly when turned on: they gradually increase in energy usage over a period of time. Ramp duration is used to simulate ramping up (power on) and down (power down) of energy that occurs in street lighting for each sunrise - sunset time. These ramps are centered on the sunrise or sunset time for each day. Specify the time it takes to power up and down for your profile. The default is 900 seconds per interval.
- **DST Participant**: Check to indicate whether your profile participates in DST. If your location does not observe Daylight Savings Time changes, uncheck this checkbox.

4. Click **Create Profile**.

   - If you are importing the file into a database, the job will be submitted to X110 using a predefined COMMMON\DATA defaulted env file. No AXDB processing will take place and the cut will always be loaded "Valid Replace".

     The env file used is the default TGX11SP.ENV:

     ```
     LOA VAL REP
     ```

   - If you are creating an LSE file, once the file is created it can be imported via Direct Input or uploaded to Interval Data Manager (IDM).

     Note: The output report will contain a reference to an LSE file that is created in the \\LODESTAR111\data\X110 folder.

     The filename format of the LSE file is as follows:

     ```
     SUNPROFILE_yyyy-MM-DD-hhmmss.LSE
     ```

Using the Data Browser

The Data Browser provides an interactive query interface that allows users to perform quick queries against the database and page through the results. The Data Browser is accessible via the **Data Browser** button from the Toolbar:

Data Browser Query

The Query tab provides a query interface for the user.

Query Options used on the Query tab include the following:

- **Database** (drop-down list): Used to select the database for the query. This can be CLDB, ELDB, ALDB, etc, based on the databases the user has access to.
- **Page Size**: Used to set the number of records to display per page in the results. The minimum is 20, maximum records per page is 999. Default is 128.
- **Select All**: Used to select (or de-select) all fields for display in results or not.
- **Load**: Used to load a previously saved query state from an xml file. The default location for query files is the **Lodestar111/Client/Data/DataBrower** folder on the client.
- **Save**: Used to save the current query to an xml file. The default location to query files is the **Lodestar111/Client/Data/DataBrower** folder on the client.
- **Reset**: Used to resets all the input fields to their initial state.
- **Cuts / Cut Series** (drop-down list): Used to specify if the query should return individual cuts or cut series.
  - When "Cuts" is selected, resulting records will be returned as individual cuts.
  - When "Cut Series" is selected, results will be grouped by CUSTID and CHANNEL. In addition: if selected, following fields will be grouped accordingly:
    - **STARTTIME**: The minimum or earliest start-time value will be returned.
    - **STOPTIME**: The maximum or latest stop-time value will be returned.
    - **STARTREADING**: The minimum value will be returned.
    - **STOPREADING**: The maximum value will be returned.

The **Query Fields** section is used to specify the query fields to be retrieved and displayed as well as the criteria required.

- **Display**: Checkboxes used to specify fields to be retrieved and displayed. Select the checkbox for the corresponding field to indicate the field should be retrieved and displayed in the results. Note that some fields can be used for the query but are not retrieved or displayed (such as INTERVALCOUNT and EDITTEXT).
- **Field Name:** List of all the field names available for query.
- **Comparison**: Drop-down list containing a list of comparison operators available to construct the criteria for the query. If left blank, this indicates there is no criterion for the field. Available compare operators include

Comparison Operator	Description
=	Field must be equal to the value specified.
>=	Field must be greater than or equal to the value specified.
<=	Field must be less than or equal to the value specified.

Comparison Operator	Description
=	Field must be equal to the value specified.
!=	Field must not be equal to the value specified.
>	Field must be greater than the value specified.
<	Field must be less than the value specified.

LIKE	Wild-card comparison. Available wild-card patterns include:
	• **%**: match any string of any length (including zero length)
	• _: match on a single character
	• Escape Character: Use "\" to escape wild card characters.
	Examples:
	`CUSTID LIKE RES%`
	Returns CUSTID that begins with 'RES'.
	`CUSTID LIKE N_23`
	Returns N1723, but not N1724.
	`CUSTID LIKE RES\_TOTL%`
	Returns CUSTID that begins with 'RES_TOTL'.

NOT LIKE	Inverse of LIKE

IN	Equal to a list of values. List must be comma separated. For regions where comma is used as decimal digit, numeric values must be enclosed in single quotes.
	Examples:
	`CHANNEL IN 1, 2`
	`METERMULTIPLIER IN '1,02', '2,014'`

NOT IN	Inverse of IN.

- **Value**: The value to compare the field to. Values can be of type string, numeric or Boolean. Boolean values must be 'Y', or 'N' values.

The **Keys List Filter** box is used to define an optional list of keys to filter the results further to a specific list of cuts. The list of keys can be full (customer-id, channel, start) or partial (customer-id, channel). Click the **Import** link to import a list of keys from existing control or text file.

Data Browser Results

The **Results** tab displays the status of the submitted query and records retrieved. It also provides options for selecting records for export.

The **Results List** window displays the records retrieved by the query. In addition, the **Results** tab includes the following options:

- **Status**: Displays the current status of the submitted query.
- **Stop**: Stops a query in progress. Status will show as "Cancelled" when complete.
- **Total Pages**: The total number of pages of records returned from the query. The number of records per page is determined by the **Page Size** setting in the **Query** tab.
- **Total Records**: The total number of records found.
- **Page**: Displays the current page of retrieved records. Use the up / down arrow or enter a page number to navigate through the pages of results.
- **Select All**: Click to select all records in the current page.
- **Select columns for copy**: Click this option and use CTRL + CLICK to select columns to be selected for copy or save.

Records can also be saved to a file. See **Data Browser Export File Formats** (below) for available output formats. The Results tab includes the following **Save Options**:

- **Save All**: Saves all pages and all columns to a file.
- **Save Selected**: Saves only the current selected records and columns to a file.
- **Copy Selected to Clipboard**: Copies the selected records and columns to the clipboard which can be pasted into other text applications such as Notepad.

Data Browser Export File Formats

Data Browser can save records into the following file formats:

- **Control File Format** (*.CTL): Control file friendly format. Fields are space-delimited and are not enclosed in quotes.
- **Comma Delimited Format** (*.CSV): Spreadsheet friendly format. Fields are comma-delimited, date and text fields are enclosed in double quotes.
- **Semi-colon Delimited Format** (*.CSV): Friendly format for locales where comma is used as decimal separator. Fields are semi-colon-delimited, date and text fields are enclosed in double quotes.

Defining Holidays, Time Of Use Periods, Seasons, and User Days in the Oracle Utilities Data Repository

Oracle Utilities Load Analysis can access data stored in relational tables in the Oracle Utilities Data Repository. This includes interval data as well as the following data:

- **Holiday Lists and Holidays**
- **Time of Use Periods and Schedules**
- **Season Periods and Schedules**
- **User-Specified Day Lists and User-Specified Days**

This section describes how to set up this data in the Oracle Utilities Data Repository, and how to have Oracle Utilities Load Analysis use this data.

Records in these tables should be created using the Data Navigator application, available through the Oracle Utilities Energy Information Platform web user interface. Refer to the *Oracle Utilities Energy Information Platform User's Guide* for more information about using Data Navigator.

Holiday Lists and Holidays

Holiday Lists and holidays are defined in the Holiday List and Holiday tables in the Oracle Utilities Data Repository.

Holiday Lists

Holiday lists are user-defined lists of holidays. For each holiday list you wish to define, create a record in the Holiday List table. Records in this table contain the following fields:

- **Name**: the name of the holiday list.
- **Note**: an optional note regarding the holiday list.

Holidays

For each holiday you wish to define, create a record in the Holiday table. Records in this table contain the following fields:

- **Holiday List**: the Holiday List (from the Holiday List table) to which the holiday belongs.
- **Date**: the date on which the holiday is observed.
- **Name**: the name of the holiday.

Holiday File Configuration File

The Holiday File configuration file is a *.cfg file stored in the C:\LODESTAR\<VERSION>\Binw directory (on client machines) or the C:\LODESTAR\<VERSION>\Common\Data directory (on the server) that specifies the holiday lists available. The format of this file is:

```
RDB/<HOLIDAY_LIST_NAME>
```

where:

- **<HOLIDAY_LIST_NAME>** is the name of a holiday list from the Holiday List table.

The Holiday File configuration file can include multiple entries, one for each holiday list.

You select the Holiday File configuration file on the Configuration tab of the Setup dialog.

Example

Below is a sample Holiday File configuration file.

```
RDB/2003_Holidays
RDB/2004_Holidays
RDB/2005_Holidays
```

Time of Use Periods and Schedules

Time of use (TOU) periods and schedules define periods of use (On Peak, Off Peak, etc.) used with Oracle Utilities Load Analysiss programs. Time of Use Periods and Schedules are defined in the LS Time of Use Schedule and LS Time of Use Period tables in the Oracle Utilities Data Repository.

LS Time of Use Schedule

For each time of use schedule you wish to define, create a record in the LS Time of Use Schedule table. Records in this table contain the following fields:

- **TOU Schedule Number**: a number that represents the TOU schedule.
- **Note**: an optional note regarding the TOU schedule.

LS Time of Use Period

For each time of use period you wish to define, create a record in the LS Time of Use Period table. Records in this table contain the following fields:

- **LS Time of Use Schedule**: the TOU schedule (from the LS Time of Use Schedule table) to which the period belongs.
- **TOU Period Number**: a number that represents the TOU period.
- **TOU Period Name**: a name for the TOU period (On Peak, Off Peak, etc.)
- **Day List**: a list of numbers representing the days to which the TOU period applies, based on the following:
  - Sunday = 1
  - Monday = 2
  - Tuesday = 3
  - Wednesday = 4
  - Thursday = 5
  - Friday = 6
  - Saturday = 7

- Holiday = 8

Example: Monday through Friday would be represented as "23456." Saturdays, Sundays, and Holidays would be represented as "178."

- **Start Time**: the start time for the TOU period.
- **Stop Time**: the stop time for the TOU period.

TOU File Configuration File

The TOU configuration file is a *.cfg file stored in the C:\LODESTAR\<VERSION>\Binw directory (on client machines) or the C:\LODESTAR\<VERSION>\Common\Data directory (on the server) that specifies that the Oracle Utilities Load Analysis programs should use the Time of Use periods and Schedules defined in the Oracle Utilities Data Repository. This file contains only a single keyword:

```
RDB/
```

You select the TOU configuration file on the Configuration tab of the Setup dialog.

Season Periods and Schedules

Season periods and schedules define seasons (Summer, Winter, Spring, Fall, or others) for use with Oracle Utilities Load Analysis programs. Seasons are defined in the LS Season Schedule, LS Season Period, and LS Season Dates tables in the Oracle Utilities Data Repository.

LS Season Schedule

For each season schedule you wish to define, create a record in the LS Season Schedule table. Records in this table contain the following fields:

- **Season Schedule Number**: a number that represents the season schedule.
- **Note**: an optional note regarding the season schedule.

LS Season Period

For each season period you wish to define, create a record in the LS Season Period table. Records in this table contain the following fields:

- **LS Season Schedule**: the season schedule (from the LS Season Schedule table) to which the period belongs.
- **LS Time of Use Schedule**: the TOU schedule (from the LS Time of Use Schedule table) to which the period belongs.
- **Season Period Number**: a number that represents the season period.
- **Season Period Name**: a name for the season period (Winter, Summer, etc.)

LS Season Dates

Records in the LS Season Dates table specify the dates for each season period. Records in this table contain the following fields:

- **LS Season Period**: the season period (from the LS Season Period table) to which the dates will apply.
- **Season Start Time**: the start time for the season period.
- **Season Stop Time**: the stop time for the season period.

Season File Configuration File

The Season configuration file is a *.cfg file stored in the C:\LODESTAR\<VERSION>\Binw directory (on client machines) or the C:\LODESTAR\<VERSION>\Common\Data directory (on the server) that specifies that the Oracle Utilities Load Analysis programs should use the Season periods and Schedules defined in the Oracle Utilities Data Repository. This file contains only a single keyword:

```
RDB/
```

You select the Season configuration file on the Advanced tab of the Setup dialog.

User-Specified Day Lists and User-Specified Days

User-Specified Day Lists and user-specified Days are defined in the User-Specified Day List and User-Specified Day tables in the Oracle Utilities Data Repository.

User-Specified Day Lists

User-Specified Day lists are user-defined lists of user-specified days. For each list you wish to define, create a record in the User-Specified Day List table. Records in this table contain the following fields:

- **Name**: the name of the list.
- **Note**: an optional note regarding the list.

User-Specified Days

For each user-specified day you wish to define, create a record in the User-Specified Day table. Records in this table contain the following fields:

- **User-Specified Day List**: the User-Specified Day List (from the User-Specified Day List table) to which the day belongs.
- **Date**: the date of the user-specified day.
- **User Day Label 1**: a name for the user-specified day.
- **User Day Label 2**: a second name for the user-specified day.

UserDay File Configuration File

The UserDay configuration file is a *.cfg file stored in the C:\LODESTAR\<VERSION>\Binw directory (on client machines) or the C:\LODESTAR\<VERSION>\Common\Data directory (on the server) that specifies the user-specified day lists available. The format of this file is:

```
RDB/<DAY_LIST_NAME>
```

where:

- **<DAY_LIST_NAME>** is the name of a list from the a name for the User-Specified Day List table.

The UserDay configuration file can include multiple entries, one for each list.

You select the UserDay configuration file on the Configuration tab of the Setup dialog.

Example

Below is a sample UserDay configuration file.

```
RDB/2004_LJP_Days
RDB/2004_RJC_Days
RDB/2004_THO_Days
```

Restrictions

The **Oracle Utilities Load Analysis Restrictions** window (**Tools->Restrictions**) displays a list of the access privileges that the System Administrator has granted to your logon ID. Specifically, it shows which programs and databases you can and cannot use. This information is view-only; you cannot modify it here.

Chapter Three

Working with Jobs

This chapter explains how to use Oracle Utilities Load Analysis to "submit a job" — that is, to select the desired Oracle Utilities Load Analysis application program, specify the necessary inputs, run the job, and view the results. It also includes a brief overview of the tools Oracle Utilities Load Analysis provides to work with your results; e.g., to view, edit, export, graph, delete, and more.

This chapter includes:

- **Submitting a Job**
- **Viewing and Managing Job Results**
- **Deleting Results**

Submitting a Job

The process of submitting a job consists of just a few steps:

- **Logging on to the Server**
- **Selecting a Program**
- **Specifying Desired Inputs and Outputs**
- **Running a Procedure**

Important Note to New Oracle Utilities Load Analysis Users: If you have never worked with Oracle Utilities Load Analysis before, be sure to refer also to the Oracle Utilities Load Analysis User's Manual that describes the program with which you wish to work. Those documents provide useful background information as well as the specific instructions you'll need to prepare the required inputs and evaluate the results for all programs in the Oracle Utilities Load Analysis system. The table in **Appendix A**: **Programs Available via Oracle Utilities Load Analysis** directs you to the appropriate reference for each program available in Oracle Utilities Load Analysis.

Logging on to the Server

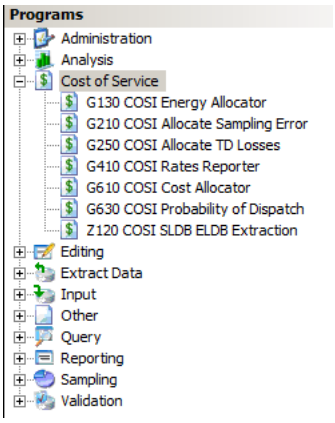
Begin at the Client Server Oracle Utilities Load Analysis desktop (shown below).

1. If you are not already logged on to the server, click the **Logon** button. For more information about logging on to the server, see **Chapter Two**: **Getting Started**.

Logged Off:  Logged On: 

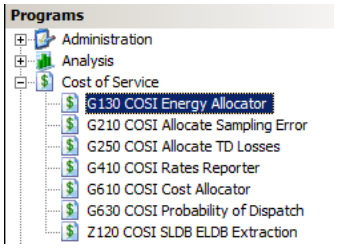
Selecting a Program

1. Select a program from the **Programs Panel** on the left by clicking on it:



When you make your selection, the programs belonging to that category appear in tree format below it. (A complete list of programs available via the Oracle Utilities Load Analysis interface is provided in **Appendix A**: **Programs Available via Oracle Utilities Load Analysis** for your convenience.)

2. Select a program.



Note: When you make your selection, the fields in the Submit Panel are automatically updated to match the type of inputs. For example, if you select an analysis program the Submit Panel will contain a button that enables you to use the Repeater to run the program multiple times. See **Chapter 7: The Repeater** for more information about the Repeater.

Customizing the Programs Panel

You can save a list of frequently used or "favorite" programs and hide/display program IDs on the Programs panel.

To save a program to your Favorites list, right-click and select "Add to Favorites."

To view your saved favorites, right-click and select "Show Favorites." The Programs panel displays a list of saved favorite programs. To remove a program from your favorites list, right-click and select "Remove from Favorites:" To restore the Programs panel to the default list of programs, right-click and deselect "Show Favorites" again.

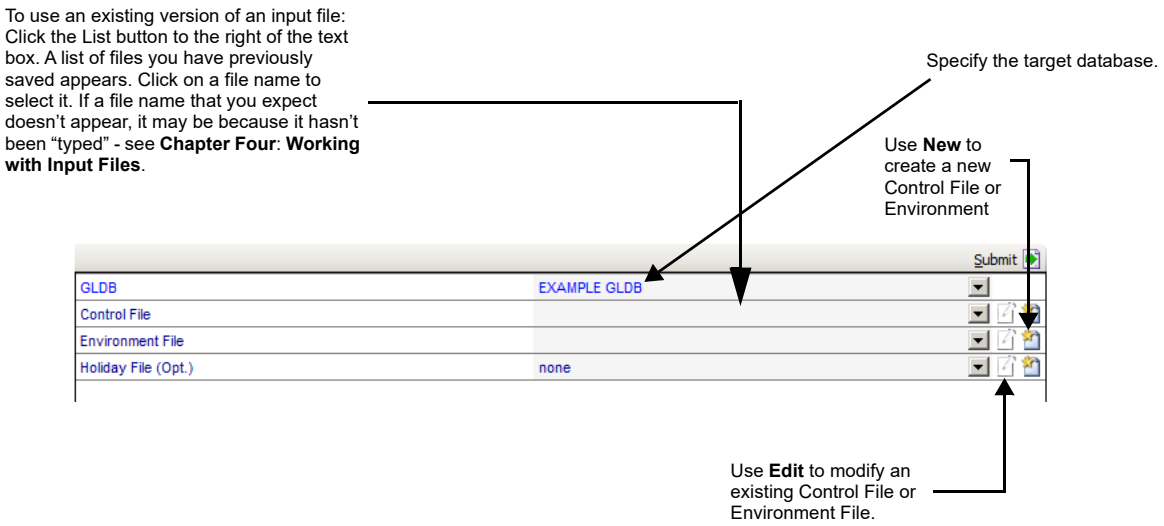
To hide the program IDs (X110, etc.) right-click and select "Hide Program ID." To restore the program IDs, right-click and deselect "Hide Program ID."

Specifying Desired Inputs and Outputs

The fields displayed on the Submit Panel vary from program to program. In general, there are fields to specify required or optional inputs, and occasionally to assign names to output files. (For example, in the illustration below, the Standard Load Analysis Program requires a Control File, an Environment File, and a version of the ELDB. It also gives you the opportunity to assign a name to the Individual Customer Statistics File and Key List File produced by the program at the end of its run.) There may also be a panel hidden underneath, accessed by clicking on the page ear, in which you can specify optional "Common files," e.g., Time-Of-Use and Holiday files.

The purpose and contents of each input and output file are beyond the scope of this document. The box at the right provides a brief definition of each file type to help orient new users. See the *Oracle Utilities Load Analysis Load Data Guides* for more information. ***Note:*** *Appendix A: Programs Available via Oracle Utilities Load Analysis lists the specific Oracle Utilities Load Analysis manual that you must refer to for detailed information about each program's inputs and outputs.*

The illustration below shows the types of elements you will encounter on a typical **Submit Panel**.



Note: Databases are listed alphabetically within each database type, in the following order:

- AXDB
- CLDB
- ALDB
- ELDB
- SLDB
- GLDB
- RLDB

To disable this option and display databases in the same order as on the Server Configuration - Databases dialog, include the following in the your ini file:

```
SortDBNames=NO
```

In general, most Oracle Utilities Load Analysis programs require you to specify three types of input files and a database:

- **Control File:** a list of the specific data records that you want to apply the program to. Typical Control Files consists of the keys of load data records or statistics records.
- **Environment File:** a set of commands that define how you want the program to process the data.
- **Database (e.g., ELDB):** contains the load data or statistics that you want to report, analyze, etc. The Load Data Analysis programs in Oracle Utilities Load Analysis work primarily with the **ELDB** (Extracted Load Database). The ELDB typically contains load data that has been checked for completeness and reliability, and is considered ready for analysis. You may have multiple ELDBs for different departments and/or applications. The Load Data Management programs work primarily with the **CLDB** (Current Load Database), which contains customer load data that is in the process of being verified. The **ALDB** and **SLDB** are archived subsets of the CLDB and ELDB, respectively. The **GLDB** and **RLDB** are utilized by the Cost of Service Interface System (COSI) optional extension to Oracle Utilities Load Analysis. For more information, see the *Cost of Service Interface (Electric) User's Guide*.
- **Common Files:** a set of files that define various schedules applied by many of the Oracle Utilities Load Analysis analysis and reporting programs. Unlike the Control and Environment files that you create for your own use and store in your personal directories, the Common Files reflect the policies of your facility and are generally used by all Oracle Utilities Load Analysis users. For that reason, they are seldom modified. They are:
  - *Holiday File* - a list of all national and local holidays observed in your service territory.
  - *Time-Of-Use Schedule File* - a schedule of on-peak, off-peak, and shoulder periods.
  - *Season File* - a schedule applying different TOU schedules to seasons throughout the year.
  - *Demand Period File* - a schedule of "on-demand" periods.
  - *User-Specified Day File* - a schedule of user-defined days to be averaged together in an analysis.
  - *Billing Cycle File* - a schedule of billing cycles for the Billed Energy program.
  - *Peak Days File* - a schedule of Peak Days for the Daytype Analysis program.

See the *Oracle Utilities Load Analysis Introductory Guides, Volumes I* and *II,* for more details.

Running a Procedure

1. Once you have completed your selections for inputs and outputs, click the **Submit** button.

   The **Job Parameters** window appears.

2. Complete the **Job Parameters** window as desired:

   - **Job Name:** This is automatically supplied (the program name plus a sequential number), but you can change it as desired. The maximum number of characters allowed for job names is 40.
   - **Start Date and Time:** *Optional*. Specify a date and time here if you don't want to run the program immediately. This will cause the execution of the program to be placed in a queue. The queue can hold up to 100 jobs. The hh:mm you specify for the time should be in the 24-hour, military time format.

   When you click **OK**, the job begins running which you can see in the **Queue** panel at the bottom of the screen.

   The **Queue** panel displays the status of all pending Oracle Utilities Load Analysis jobs. Depending upon how the System Administrator has configured the system, you may see all jobs that are waiting in the queue, or just those that belong to you. The list in the window is automatically updated by default every eight seconds. When a job has completed successfully, the system marks it **Done** and, a few seconds later, clears it from the screen if auto clear is selected in the setup screen.

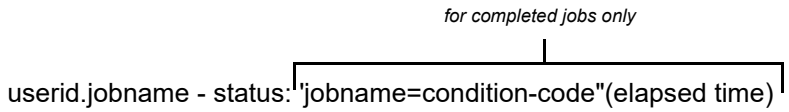
   Cancel Job *button*: To cancel a job, highlight its entry in the **Queue** panel and click the **Cancel Job** button.

   Refresh *button*: To refresh all jobs from the Queue panel, click the **Refresh Jobs** button.

   Clear Queue *button*: To clear all jobs from the Queue panel, click the **Clear Queue** button.

   Results *button*: To refresh all jobs in the **Results** panel, click the **Results** button.

   About Status messages: Status messages in the **Queue** panel follow this format:

   Example*:*

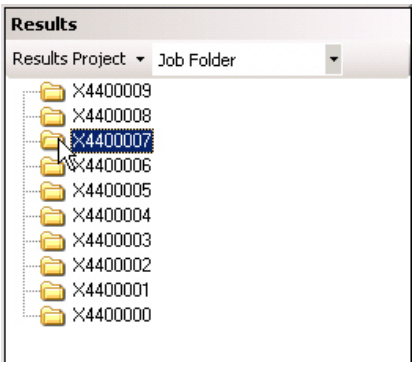
   UID.Y3104079 - Done: 'UID.Y3104079=99'(0 min, 5 sec)

   - **Userid**: Your network logon ID.
   - **Jobname**: Name you or the system assigned in the **Job Parameters** window.
   - **Status**: Current status and, if the job is waiting in the queue, its Oracle Utilities Load Analysis program ID and the time that it's scheduled to run.
   - **Condition Code**: Condition codes tell you whether or not the job completed successfully. A "0" code means successful completion. A condition code of anything other than 0 may indicate a problem. A "99" code, for example, means that Oracle Utilities Load Analysis detected an error, possibly in your Environment File or Control File, and that the job terminated abnormally. ***If you see a condition code of anything other than 0, be sure to check the output reports for error messages (see next section).***
   - **Elapsed Time***:* Time between the start of the job and its completion.

Viewing and Managing Job Results

How to View and Edit Results:

1. To view the results of a completed job, click within **Results** panel on the right of the screen.



2. Highlight the desired job directory in the Results list.

   A list of files for the selected job appears in the panel below.

   You can delete, copy, and move job directories by selecting a job and clicking the right mouse button.

   - To delete a job directory, select **Delete**.
   - To move a job directory, select **Move** and select the destination to which you wish to move the directory.
   - To copy a job directory, select **Copy** and select the destination to which you wish to copy the directory.

3. Select the file you wish to view.

   The following types of files are generally produced at the end of a successful job run:

   - **cslstar.job:** list of Environment File variables used and their values.
   - **dbparms.inf**: database connection information. Listed only when working with data stored in relational tables
   - **report.html:** text file containing all reports produced by the job run. Oracle Utilities Load Analysis automatically places a useful report index at the end of this file.

     Note: Programs X110, and X111 include a "Combine Reports" checkbox that if checked (default) combines all reports at the end of the job (as in previous versions). If this option is disabled (unchecked), the reports are output separately. The names of the separated reports will follow the naming convention described below:

     <ProgramID>Sysprint<n>.rep when multiple runs of the same program are invoked

     For example, CLDB Production input (X111) with Auto Editor engaged will produce the following reports:

     - X111Report.html - This will be the Direct Input program report.
     - X210Report.html - This will be the initial Validation Report.
     - X310Report.html - This is the Editor Report.
     - X210Report.html - This is the validation report of the cuts that have been edited.
     - X410Report.html - This is the reporter step for the cuts that did not pass validation.

- **report.html:** an html file containing all reports produced by the job run. Oracle Utilities Load Analysis automatically places a useful report index at the end of this file. See **Viewing the REPORT.HTML, SYSGRAPH.HTM, and SYSGRAPH_STR.HTM Files** on page 3-10 for more information about viewing this file.
- **sysgraph.htm:** an html file that allows users to view graphs of the cuts produced by the job run. See **Viewing the REPORT.HTML, SYSGRAPH.HTM, and SYSGRAPH_STR.HTM Files** on page 3-10 for more information about viewing this file.
- **sysgraph_str.htm:** html files (sysgraph_str1.htm, sysgraph_str2.htm, and so on) that allow users to view graphs of the cuts for each stratum defined in the job run. See **Viewing the REPORT.HTML, SYSGRAPH.HTM, and SYSGRAPH_STR.HTM Files** on page 3-10 for more information about viewing this file.
- **(name).dat:** file containing data generated by the program, such as individual customer statistics.
- **run.log:** text file of information about the job run, such as processing time. This is useful for diagnostic purposes if there is a problem with the run.
- **<FILE_NAME>.XML** (Key Generator only): XML file containing results of a key generator query, where <FILE_NAME> is the name of the Output Keys File specified on the Key Generator Submit screen. See **Creating XML Control File Query Lists** on page 13-39 in Oracle Utilities Load Analysis Load Data Management User's Guide for more information about creating XML output from the Key Generator.

  Note: The control file composer for the Key Generator will allow mixed-case to support XML.

In addition, Oracle Utilities Load Analysis puts a copy of your input files in the job directory.

4. Select a tool for viewing or editing the highlighted file. The **Results** window gives you access to several tools for viewing and manipulating the contents of a selected file:
   - **Right Click Options**: In addition, you can click the right mouse button and select from a number of options, including:
     - **Copy**: Allows copying the file to another directory. See **Copying Results to Another Directory** on page 3-9 for more information.
     - **Delete**: Allows you to delete the selected file.
     - **Rename**: Allows you to rename the selected file.
     - **Load XML as List**: Loads XML output created by the Key Generator as a list in the CLSLSLISTS.MDB database. Once loaded, XML lists can be used when creating control files.
     - **Generate Sequencer Step Command**: Generates a sequencer step template for this job. This is useful for assisting in the creation of a SEQ file. See **Generating Sequencer Step Commands from Job Results** on page 6-5 for details about enabling this option.
     - **Open With**: Allows you to open the selected file with the Default Viewer, Alternate Viewer or an Associated Program (defined on the Configuration tab of the Setup dialog).
     - **Show System Files:** Allows you to view the system files such as *.log and *.CFG that are listed above under item 3.

Copying Results to Another Directory

You can copy any of your job results files to desired directories without having to leave the **Results** window.

How to Copy Results:

1. In the **Results** window, highlight the job file you wish to copy.
2. With your cursor in the Job Files List window, click the *right* mouse button, then select **Copy** from the drop-down list that appears. Select from the following options:
   - **Copy to Production Input Directory:** Copies the selected file to the \COMMON\LOAD directory.
   - **Copy to Common Files Directory:** Copies the selected file to the \COMMON\DATA directory.
   - **Copy to Client Data Directory:** Copies the selected file to the DATA directory on your local hard drive.
   - **Copy to Sequencer Input Directory:** Copies the selected file to the sequencer input folder. See **Copying Files to the Sequencer Input Folder** on page 6-4 for details about enabling this option.
   - **Copy to Anywhere:** Copies the file to a directory you specify.

Managing Results with Project Folders

You can use project folders to organize your job results. By default, the project folders you create are located in the Common/Data/Results directory. You can change the location of this directory by specifying a new file path. See **File Paths** on page 2-6 for more information on changing file paths.

How to Create a New Project Folder

1. Click **Results Project** and select **New Results Project.**
2. Enter a name for the project and click **OK.**

   The new project folder is created in the Results directory and appears in the results project list.

How to Rename or Delete Project Folder

1. In the results project list, select the project folder that you want to rename or delete.
2. Click **Results Project** and select from the following options:

   Rename Results Project: Allows you to rename the project folder.

   Delete Results Project: Allows you to delete the project folder and its contents.

How to Copy or Move Folders to a Project Folder

1. Select one or more folders in the Results panel.
2. Right click and select from the following options:

   Copy to Project: Allows you to copy the project folder and its contents to another project folder. If a folder already exists, you will be prompted to overwrite the contents in that folder.

   Move to Project: Allows you to move the project folder and its contents to another project folder.

Renaming Results Files

It is also possible to assign a new name to a file. Highlight the desired file under **Job Files** in the **Results** window. Click the *right* mouse button, then select **Rename** from the drop-down list that appears.

Viewing the REPORT.HTML, SYSGRAPH.HTM, and SYSGRAPH_STR.HTM Files

The REPORT.HTML file is an HTML file that allows users to view the reports in a Web browser such as Microsoft Internet Explorer. The SYSGRAPH.HTM file is an HTML file that allows users to view graphs of the cuts produced by the job run in a web browser such as Microsoft Internet Explorer. The SYSGRAPH_STR.HTM files are additional HTML files (SYSGRAPH_STR1, SYSGRAPH_STR2, and so on) produced by the Analysis programs that allow users to view graphs for each stratum defined in the job run.

Viewing the REPORT.HTML File

To view the REPORT.HTML file in Internet Explorer, select the REPORT.HTML file, then click the right mouse button and select **Open With-›Associated Program.** Or you can double click on it if Internet Explorer is your default browser. The file opens in an Internet Explorer window.

> **Note:** There is a setting in the **Tools** tab of the **Setup** window which warns users if they are about to view a report that is very large, since if a report exceeds memory limitations it can cause the browser to freeze. The default is 10 MB and a popup a message will appear allowing the user to cancel the load if the file is larger than the preset limit.

You can expand or collapse a report by clicking on the report title.

The following options are available for each report under **Options**:

- **Expand All:** Expands all of the reports in the file.
- **Collapse All:** Collapses all of the reports in the file.
- **Hide All But This:** Hides all reports except for the selected report. You can use this option if you want to print a single report from your Web browser.
- **Show All:** Displays all of the reports in the file if they are hidden.

Viewing the SYSGRAPH.HTM and SYSGRAPH_STR.HTM Files

To view the SYSGRAPH.HTM or SYSGRAPH_STR.HTM file in Internet Explorer, select the file, click the right mouse button and select **Open With->Associated Program** or double click on it if Internet Explorer is your default browser. The file opens in an Internet Explorer window.

The window displays a list of cuts produced by the job run. For each cut the window displays the following:

- **Number**: The number of the cut. This is used for informational purposes on this screen only.
- **Graph?** (checkbox): A checkbox that indicates if the cut should be included in the graph display.
- **Recorder ID**: The recorder ID of the cut.
- **Channel.**: The channel number of the cut.
- **Stratum**: The stratum to which the cut is assigned. This value is displayed in the SYSGRAPH_STR.HTM files.
- **Start**: The start time of the cut.
- **Stop**: The stop time of the cut.
- **SPI**: The seconds-per-interval (SPI) of the cut.
- **UOM**: The unit-of-measure (UOM) of the cut.
- **Total**: The total of the interval values in the cut.
- **Load Factor**: The calculated load factor of the cut. This is displayed in the SYSGRAPH_STR.HTM files.
- **Demand (Non-coincident)**: The peak interval value of the cut. This value is displayed in the SYSGRAPH_STR.HTM files.
- **Demand (Coincident)**: The interval value of the cut at the time of coincident peak. This value is displayed in the SYSGRAPH_STR.HTM files. Cuts with no coincident value will display "NaN".
- **Plot Type**: The plot type for graphs for the cut. This value are two options: Time Series and Duration.
  - **Time Series**: The X-axis displays the Start Time through the Stop Time of the cut.
  - **Duration**: The cut's intervals are sorted in descending order and the X-axis is based on the interval count starting from 1 through the last interval in the cut.

  Note: When plotting graphs in combined graph mode (overlay or stack), all cuts must be designated with the same plot type.
- **Color**: The custom color associated with the cut. This column only appears if the Custom option is selected from the Color drop-down list. To select the custom color for the cut, click the colored square and select the desired color from the dialog.

The Graphing Toolbar

The graphing toolbar contains the following:

Display Graph | Cuts: Toggles switching of the current view between displaying graph view or displaying cuts view. Note that at least 1 cut must be selected in order to display graph.

Options: Opens the **Graphing Options** dialog, containing the following graphing options:

- **Precision**: Controls the number of significant digits of the Y-Axis labels. The default is 4.
- **Major Tick Marks**: To set the default number of major tick marks along the X-axis for the graphs, select the desired number from the **Major Tick Marks** drop-down list.

- **Minor Tick Marks**: To set the default number of minor tick marks along the X-axis for the graphs, select the desired number from the **Minor Tick Marks** drop-down list.
- **Default Color**: To set a default custom color, click the **Custom Default Color** colored square and select the desired color from the dialog.
- **Horizontal Grid Lines**: To set a horizontal grid line at a specified value, enter that value in the **Custom Grid Lines** box. For example, custom grid line can be set at 0 for better visualization of the positive and negative values on the graph. Grid lines will not be drawn if the provided vale is outside the scope of the graph.
- **Stratum Breakpoints**: Values and colors for stratum breakpoints (used with Y310 and Y330).
- **Date Format**: The date format used on the graph. Options include mm/dd/yyyy and dd/mm/yyyy/
- **Title**: For Single and Stack graphs (see below), you can specify a default title for graphs by entering the title in the **Graph Title** box. Note that this option is not available for Individual graphs.
- Click **OK** to save your settings. Click **Reset** to restore the settings to their previous values.

Graph Type: Select the type of graph to display. There are five graph types:

- **Line Chart - Separate**: Displays each graphed cut as a separate graph.
- **Line Chart - Standard**: Displays each graphed cut on the same graph, with each cut overlaying each other.
- **Line Chart - Stacked**: Displays each graphed cut on the same graph, with each cut stacked one on top each other.
- **Bar Chart**: Displays each graphed cut as a bar in a bar graph.
- **Scatterplot**: Displays each graphed cut as a point on a scatterplot graph.

Sorted: Indicates that the bar graph data is displayed in descending order. This is displayed only on Bar graphs.

Data: Sets the type of data which is displayed on the bar graph. Default value is **Total.** This is displayed only on Bar graphs.

X Value: Sets the type of data plotted on the horizontal axis of the scatterplot graph. Default value is **Total.** This is displayed only on Scatterplot graphs.

Y Value: Sets the type of data plotted on the vertical axis of the scatterplot graph. Default value is **Demand (Non-coincident).** This is displayed only on Scatterplot graphs.

Size: Set the default graphing size (auto, 50%, 100%, 125%). Select the desired size from the Graph Size drop-down list. If auto (default) is selected, the size of the graph is determined by the cut.

Color: The colors of the lines on the graph can be either set automatically or be customized. To specify which setting you prefer, select the desired setting (auto or manual) from the Colors drop-down list.

Highlight Status Codes: To highlight status codes in the graph(s), select the desired comparison ("None", "Equal To", or "Worse Than or Equal To") from the Highlight Status Codes drop-down list, and the specific status code from the status code drop-down list.

Track: When track is enabled, an interactive line is displayed when the mouse cursor moves over the graph. Clicking on the graph inserts a line in the graph. Up to two lines can be in place at one time. This is displayed only on Line and Scatterplot graphs.

Zoom: When two track lines are placed on a graph, the Zoom option is enabled. Click **Zoom** to rescale the graph between the lines. This is displayed only on Line and Scatterplot graphs.

Viewing Graphs

To view a graph of one or more of the listed cuts, select the checkbox that corresponds to the cut(s) you wish to view and click the **Graph** button. The selected cuts appear on a graph in the window.

Note: For Single and Stacked graphs, all selected cuts must have the same SPI value.

The graphs display the following information:

- **Data Panel:** The side panel displays information for the cuts displayed in the graph. Using the Data Panel, you can:
    - Select a cut to highlight the cut in the graph. Multiple cuts can be selected by holding down the Ctrl key when selecting cuts. In addition, a series of cuts can be selected by holding down the Shift key and selecting the first then last cut in series.
    - Use the **Filter** menu to display the selected cuts in the graph, or to change the graph type for the current selection. After filtering the graph, select **Clear** to reset the graph.
    - Use the Color column on Single and Stacked graphs to change the color of the cuts displayed on the graph.

      Note: The Individual graph does not display a data panel.

Display Panel: The panel that displays the graph(s).

Individual, Single, and Stacked graphs display the following information:

- **Title**: The title of the graph. Note that this does NOT appear for Individual graphs.
- **CUST ID**: The recorder ID and channel number of the cut. Appears ONLY on Individual graphs.
- **Start-Stop**: The start time and stop time of the cut(s). If these values differ between cuts, the earliest Start Time and the latest Stop Time are displayed.
- **UOM**: The unit-of-measure (UOM) of the cut(s).

You can use the graph to:

- Display additional information by hovering over the cut.
- Highlight the data for the cut in the Data Panel by selecting the cut in the graph. Multiple cuts can be selected by holding down the Ctrl key while selecting cuts.

To return to the Cuts screen, click **Cuts** in the graphing toolbar.

Deleting Results

Oracle Utilities Load Analysis saves the results of each program that you run in a separate job directory on the server. Periodically, you should delete unwanted directories and files.

How to Delete Job Directories:

1. At the Oracle Utilities Load Analysis screen, click within **Results** panel area and highlight the desired job directory, then right-click and select **Delete**. If you wish to delete a number of directories at once, highlight each one while holding down the SHIFT key on your keyboard, then click **Delete**. Instead of clicking **Delete**, you can press the **DELETE** key on your keyboard.

   A box appears asking you to verify that you really wish to make the deletions.

2. Respond to the question as desired. If you click **Yes**, the directory and all of its contents are deleted.

You can also delete selected files within a job directory.

How to Delete Job Files:

1. In the **Results** panel highlight the job directory containing the file you wish to remove. Then, highlight the file in the **Job Files** list.

2. Click the *right* mouse button, then select **Delete** from the drop-down list that appears. Respond to the warning as desired.

Chapter Four

Working with Input Files

This chapter explains how to create and modify Environment, Control, and Common files using Oracle Utilities Load Analysis's **File Composer**. It also describes how to manage "type information" — an important link between programs and the input files you've created to use with them.

This chapter includes:

- **Overview**
- **Creating and Editing Environment and Control Files**

Overview

This section provides a brief explanation of the Oracle Utilities Load Analysis features and functions covered in this chapter, including:

- **File Composer**
- **File Types**
- **Graphical Key Generator**
- **Common Files**

File Composer

The File Composer is a "smart editor" that guides you through the process of creating, editing, and saving Environment and Control files. It is especially useful for working with Environment files, since it automatically displays the appropriate command and parameter options for each type of file.

File Types

"Type" is an important file management concept in Oracle Utilities Load Analysis. A file's type identifies the program that the Environment File or Control File works with. For example, an Environment File might be typed to work with the 100% Sample Analysis Program. In other words, type is an internal association between an input file and a program. Types are useful because they make it possible for Oracle Utilities Load Analysis to automatically display only those files or file options that are appropriate to the program you're using.

Graphical Key Generator

The **Graphical Key Generator** is used for creating or editing Simple Key Generator Control files.

Common Files

The term "Common Files" refers to the Holiday File, Time-Of-Use Schedule File, Season File, Demand Period File, Billing Cycle File, Peak Days File, and User-Specified Day File. Typically, Common files are established according to the policies of your facility, and are used by all Oracle Utilities Load Analysis users. For that reason, they are typically stored on the server, rather than your local drive. *You should not modify these files except under special circumstances.* Occasionally, you may want to try out "What If" scenarios using test versions of the Common files. In that case, it is recommended that you save the experimental versions with meaningful names that are easily recognized by you and by other users.

You'll find a description of the commands that make up the Common files in the *Oracle Utilities Load Analysis Load Data Analysis User's Guide*.

Creating and Editing Environment and Control Files

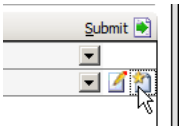
This section describes how to create and edit Environment or Control files, including:

- **Using the File Composer**
- **Creating an Environment File**
- **Creating a Control File**
- **Including Comments in Control and Environment Files**
- **Transfer Type Utility**

Using the File Composer

How to Launch the File Composer:

1. You can also open the **File Composer** by clicking the **New** or **Edit** button on a **Submit Panel**.



2. Compose the Environment File or Control File as desired.

   See **Creating an Environment File** on page 4-4 for instructions for creating an Environment File. See **Creating a Control File** on page 4-4 for instructions for creating a Control File.

   Note: The composer window includes two column number and row number indicators on the bottom of the screen, in order to indicate how many characters have already been typed. This is helpful since there is a line size character limit.

Composer Tools

The Composer window displays buttons along the top of the text box to help when creating files:

- **Sort**: sorts the settings in the file alphabetically
- **Clean**: reformats the settings in the file, cleaning and removing up any extra blank spaces in commands
- **Sum**: sums the value of a specified field
- **Group**: removes duplicate settings from the file
- **Comment**: comments out the selected settings
- **Uncomment**: uncomments the selected settings
- **Explore Input Folder**: Opens the input folder for a sequencer file. See **Copying Files to the Sequencer Input Folder** on page 6-4 for details about enabling this option.

Entering Commands in the Composer

There are four methods for using the **File Composer**, which can be used interchangeably:

- Using your keyboard, you can simply type your entries in the left list box. You can edit text in the left list box using standard text editing techniques and the **Edit** controls (**Cut**, **Paste**, **Copy**, and **Undo**).

  Note: Tab characters are not allowed in Control, Environment, and other input files.

- Select a command or parameter from the pane on the right and click **Paste**.
- Double-click a command or parameter in the pane on the right.
- *Right*-click in the pane on the left and select a command or parameter from the pop-up menu that appears. **Note**: This option is only available if the **Right Mouse Button Behavior** option is set to **Composer** on the **Tools** tab of the **Setup** dialog.

Creating an Environment File

All of the commands that pertain to the specified type of file appear in the File Composer's right list box. As you create your file by selecting commands and parameters, they appear in the left list box.

You can find detailed descriptions of the commands and parameters that make up a particular type of file by referring to the appropriate Oracle Utilities Load Analysis manual (see **Appendix A**: **Programs Available via Oracle Utilities Load Analysis**). You can select commands in the right list box in any order, but be sure to select all that are required (the system will not automatically alert you if you leave something out).

How to Create an Environment File:

1. Select a command from the list box on the right side of the File Composer, using one of the four methods described under **Creating and Editing Environment and Control Files** on page 4-3.

   The first three letters of the command will appear in the left pane.

2. Specify parameters, using one of the four methods described under **Creating and Editing Environment and Control Files** on page 4-3.

   Note: In some cases, a parameter shown in the list on the right is merely a placeholder to remind you of a parameter's format. For instance, the parameter **mmddyyhhmmss** stands for the date and time. When you paste the parameter into the left pane, you will then need to highlight and edit it. For example, 060195000000.

   The selected parameter(s) will appear in the left pane, following the command selected in Step 1.

3. Click **Next** to specify another command (repeat steps 1 and 2), as needed.

4. When you have completed the file as desired, click **Close** or **Save**, then name and save the file.

   Give your file a unique, useful name (up to 105 characters long, including spaces). You do not need to include the 3-letter file name extensions. Oracle Utilities Load Analysis supplies the file extensions by default.

   Also, save the file in the default directory, shown in the dialog box just under **Directories**. In the illustration above, it's c:\cslstar\data. (For Control and Environment files, the default directory is defined in the **File Paths** tab under **Tools->Options**.) Otherwise, the file won't be available for the **Submit Panel** (see **Chapter Three**: **Working with Jobs**).

Creating a Control File

You create the Control File by typing text into the text box. You can apply any of the **Edit** controls (**Cut**, **Copy**, **Paste**, and **Undo**).

Templates: Templates are "master" Control and Environment files that you can use as a starting point for your own versions. Working from a template saves you the effort of typing an entire new file. All **Control** and **Environment** files have appropriate templates.

> **Note**: All template files must reside in the server Common\Data folder and have a file extension of ".tem"

Oracle delivers a set of templates with the Oracle Utilities Load Analysis software (a list is provided in **Appendix C**: **Oracle Utilities-Supplied Input Files**). Oracle Utilities Load Analysis recognizes the templates by file name. The convention is TG*Xxx*A for Control files and TG*Xxx*B for Environment files, where Xxx is the program identifier (for example, TGY22A is the Control File for the program Y220, Manual Entry). If your facility wants to use a different file for a template, you must give it the file name specified in **Appendix C**: **Oracle Utilities-Supplied Input Files**. Otherwise, the file will not appear when you click the **Template** button.

When you have completed the file as desired, click **Close** or **Save**, then name and save the file. Also, save the file in the default directory, shown in the dialog box just under **Directories**. (For Control and Environment files, the default directory is defined in the **File Paths** tab under **Tools›Options**.) Otherwise, the file won't be available for the **Submit Panel** (see **Chapter Three**: **Working with Jobs**).

Using Key Generator Lists

The Key Generator can create lists in XML format that can be used when creating control files. See **Creating XML Control File Query Lists** on page 13-39 in Oracle Utilities Load Analysis Load Data Management User's Guide for more information about creating XML output from the Key Generator.

> **Note:** The control file composer for the Key Generator will allow mixed-case to support XML.

How to select values from a Key Generator list:

1. Click **List** button on the File Composer. The **Defined Lists** window opens.
2. Select the Key Generator list from which you wish to select the value from the **Your Defined Lists** drop-down list. Only lists created with your User ID or a User ID of "PUBLIC" are displayed. The selected listed appears in the window.
3. To add values from the displayed list to the control file, do the following:
   a. Select the appropriate record from the record list. To select all records, click **Select All**.
   b. Select the column that corresponds to the value you wish to select in the **Columns to insert** list. To select multiple columns, hold down the Ctrl key and select multiple items. The CUSTID and CHANNEL columns are selected by default.
   c. Click **Insert**. The values from the selected records/columns are inserted into the control file.
   d. Click **Close** to close the Defined Lists window.

Including Comments in Control and Environment Files

Comments can be included in Control and Environment files and may appear anywhere a blank is valid.

Most comments are stripped out of the Control and Environment files and replaced with a blank before being processed by programs. However, a comment that is contained within a quoted string will be retained.

The following rules apply when including comments:

1. **All** comments **must** begin with slash-asterisk (/*) and end with either asterisk-slash (*/) or the end of the line. The comment-ending asterisk-slash symbols are therefore optional and comments may not extend across lines.
2. A comment that is totally contained within either single quotes or double quotes (including its initial slash-asterisk and terminating asterisk-slash if present) will be left as it is, i.e., the comment will remain in the file.
3. Any comments not totally contained within quotes will be replaced by a single blank.
4. Any line containing nothing but comments will be deleted.
5. Input files to programs that can read comments will not have the comments removed. These include the following:
    - **Key Generator:** G810, X810, X820, Y810, Y820
    - **Transformation:** X620, Y620
    - **Invalid Series Validation:** X220
    - **Sampling** control files that use our user language

Transfer Type Utility

Oracle Utilities Load Analysis **Type Transfer Utility**, used for convert multi-typed information from one CSLSINFO.MDB to another.

You can run the Type Transfer Utility by selecting the **File->Type Transfer Utility** menu**.**

1. When the window appears you can browse for the CSLSINFO.MDB file in the **Select Info Database (CSLSINFO.MDB) to Import:** field.
2. Then select **Browse** for **Location of CTL and ENV files to Import.**

Chapter Five

Managing Data

This chapter provides some helpful tips for managing your interval data records and statistics records (both called "cuts"). Specifically, it introduces the three programs that are available for getting cuts into the interval databases, where you can apply the Oracle Utilities Load Analysis programs to them. The chapter then explains how you can delete cuts from a Oracle Utilities Load Analysis database, and how to move or copy cuts from one database to another.

This chapter includes:

- **Getting Data into the Interval Databases**
- **Moving, Copying, and Deleting Cuts**

Getting Data into the Interval Databases

The CLDB (Current Load Database) and the ELDB (Extracted Load Database) are the "working" interval databases for the Oracle Utilities Load Analysis Load Data Management and Analysis programs, respectively. That is, the data that you want to work with must be in one of these databases. You may actually have a number of CLDBs and ELDBs, depending upon the needs and policies of your facility.

Input Programs

In Oracle Utilities Load Analysis, four programs are available for getting data into an interval database. Which program you use is determined by where the data comes from:

- **Direct Input** (X110, Y130)**:** This program is used to input a relatively small file of interval data. X110 and Y130 are for .LSE format. The file must be in the Direct Input format, and it must reside on your local drive. You'll find instructions for using Direct Input in the Oracle Utilities Load Analysis Load Data User's Guides.
- **Production Input** (X111, Y131)**:** Used to input large amounts of interval data from a translator into the database on a regular, "production" basis. X111 and Y131 are for .LSE format. The only difference between Direct Input and Production Input is the location of the interval data input files. The Production Input files must reside on a network server. Because these files are usually very large and require a long time to process, you would typically schedule them to run at night or during another off-period.
- **Manual Entry** (X120, Y220)**:** Used to input a small, manually-created file that is available on your local data directory.
- **Move/Copy/Delete** (Q91M, Q91C, Q91D)**:** Used to delete cuts, or to move or copy cuts from one Oracle Utilities Load Analysis database to another. This program applies just to Oracle Utilities Load Analysis databases. You'll find instructions for using these programs on the following pages.

  Note: The Move/Copy/Delete programs support embedded SQL commands and the pre-process key generator. For additional information on this functionality, see the chapter "Oracle Utilities Load Analysis Mechanics" in the *Oracle Utilities Load Analysis Load Data Analysis User's Guide.*

Moving, Copying, and Deleting Cuts

The remainder of this chapter explains how to delete cuts from a database, and how to move or copy cuts from one database to another.

1. Select a program category from the categories displayed along the left side of the screen.
2. Select a program from the program names displayed below each category.

   Your selections depend upon which operation you wish to perform:

To do this:	***Select....***	
	Category	***Program***
Move cut(s) from one database to another, deleting the original(s)	Administrator	Move Cuts
Copy cut(s) from one database to another, leaving the original(s) intact	Retrieval	Copy Cuts
Delete cut(s) from a database	Editing	Delete Cuts

3. Specify a Control File.

   The Control File is a list of the cuts that you want to move, copy, or delete. The cuts may be interval data records or statistics records.

   You can use an existing Control File as-is or with modifications, or you can create a new one from scratch using the Oracle Utilities Load Analysis **File Composer** (described in **Chapter Four**: **Working with Input Files**), or the Key Generators. In any case, the file must conform to the format specifications below.

   Control File Format: For each cut that you want to move, copy, or delete, you must specify its *full key*, as shown below.

```
customer-id,channel-number,start-time
```

   The start-time for interval data records must be formatted as follows: `mm/dd/yy-hh:mm:ss` or `mmddyyhhmmss`.

   Delimiters (separators between the parts of the cut key) may be commas (as shown above) or blanks.

   Note: Each program run will perform just one operation (Move, Copy, or Delete) on all valid keys in the file.

   You can have an unlimited number of cut keys in your Control File.

   Examples: The sample Control File shown below identifies five interval data records:

```
N1723,1,06/01/95-00:01:00
N1723,2,06/01/95-00:01:00
N1727,1,06/01/95-00:01:00
N1743,1,07/01/95-10:57:00
N1754,1,07/04/95-11:30:00
```

   The next sample Control File identifies five statistics records:

```
RESI-NCD-SMEN,0,05/01/95-00:01:00
RESI-NGY-SMEN,0,05/01/95-00:01:00
RESI-LFC-SMEN,0,05/01/95-00:01:00
RESI-LFN-SMEN,0,05/01/95-00:01:00
RESI-DF-SMEN,0,05/01/95-00:01:00
```

4. Specify an Environment File.

   The Environment File describes how you want the cuts to be processed. You can specify an existing Environment File, or you can use the **File Composer** to create a new one or modify an old one.

   Environment File Format: The commands and their parameters that make up the Environment File are shown in the box below. Only the three letters shown in capitals are required (for example, RPL). You can enter commands and parameters in upper and/or lower case; the system will automatically force all letters to upper case. You must put each command on its own line in the file, as follows:

   RPL

   REPort [ALL | EXCeptions]

   KEYlist

   ALL

   - **RPL:** (Optional, may be used with Move Cuts or Copy Cuts) If a cut with the same key already exists in the target database, the program replaces the existing cut with the new one. ***Note:*** This operation replaces an existing cut's edit trails, too. So, if for some reason you need to recover an old cut after applying the Move Cuts or Copy Cuts program, you'll need to get it from a recent backup of the original database.

     The default for Move or Copy is to **not** replace an existing cut; that is, the existing cut is left as-is, and the new cut is **not** entered into the database.

   - **REPort:** Specifies the contents of the Execution Log produced at the end of the program run. **REP ALL** reports the status of all keys listed in the Control File. With **REP EXC**, only the cuts that failed the operation are listed. **REP ALL** is the default. In either case, the Summary Report produced at the end of the program run will indicate the total number of keys processed.

   - **KEYlist:** (Optional) Produces an output file of any keys from the Control File that the program could not find in the database.

   - **ALL:** (Optional) Copies all cuts from one database to another without the need of a Control File.

     Notes: ALL will be valid only with the COPy function. If coded in conjunction with either MOVe or DELete the program will issue an error message and terminate.

     When the COPy function is run with ALL, any Control File supplied will be ignored.

     Upon a successful run of COPy ALL, separate counts of the number of cuts, archive cuts and edit trails copied will be displayed.

5. Specify the remaining inputs and outputs.

6. Complete the **Job Parameters** window.

   See **Running a Procedure** on page 3-6 if you need additional instructions.

7. View the results.

   In the job directory produced by the program, you will find a Sysprint File and (optionally) a Keylist File. (See **Viewing and Managing Job Results** on page 3-7 for instructions on how to locate and open the job directory.)

   The Sysprint File consists of reports that let you know whether or not the job processed successfully. Be sure to check these reports carefully:

   - **Environment Report:** lists back the commands from the Environment File.

- **Execution Log:** reports the status of each cut in the Control File (i.e., whether or not the operation was successful, per cut). If you specified REP EXC in the Environment File, the report includes only cuts that were **not** successfully processed.
- **Summary Report:** contains useful statistics about the job; how many cuts were processed, how many cuts were successfully processed, and how many cuts were unsuccessful.

Important Note: If you are copying or moving cuts from a CLDB or ALDB into an ELDB or SLDB, all edit trails and archives will be removed. In addition to this, the edited flag will be changed to an "N" from a "Y". This is due to the requirement that ELDBs must not contain audit or edit information. Additionally, the following keywords apply DBA ELDB (Target database) and DBA CLDB for the (Source Database).

Chapter Six

The Program Sequencer

This chapter explains how to define and run a customized sequence of Oracle Utilities and non-Oracle Utilities programs using the Program Sequencer, and includes:

- **Overview**
- **Establishing Directories for Sequencer-Related Files**
- **The SEQ File**
- **Specifying Dates in the Sequencer**
- **Running the Sequencer via Windows Schedule Service**
- **Sequencer Security**
- **Sequencer Output**

Overview

The Program Sequencer lets you "chain" together a series of Oracle Utilities Load Analysis, user-written, and/or third party executable programs in a customized sequence. It lets you specify — in one concise Command File — instructions for executing up to 99 programs in a single job stream, including the sequence of the programs, the desired inputs for each, and the destination directory and databases/tablesets for the results.

Once you have defined the sequence, you can apply the Windows Schedule Service to have it process unattended in batch mode—automatically without additional user intervention. You could schedule it to run automatically at one future date and time, or you could schedule it for repeated periodic execution. For example, you might create a sequence that automatically loads LSE files every Wednesday and Friday at 1:00 am.

You can also submit the sequence for immediate processing using the Oracle Utilities Load Analysis Graphical User Interface.

This chapter explains how to define a program sequence and its inputs, and how to submit the job in interactive or batch mode.

Note: You cannot use the Sequencer for most of the Sampling programs. The exception is B210 which may be used to query delimited files and create control or environment files or commands. See the *Oracle Utilities Load Analysis Sampling Package User's Guide* for more information.

Establishing Directories for Sequencer-Related Files

As you will see in the following pages, you define a sequence of programs and their inputs in a Command File called a "SEQ" File (named after the '.seq' extension that identifies this type of file to the system).

The Sequencer can access input files required by the program being run in one of two ways; by searching the SEQINPFILS location (described below), or by having a full path and filename specified in the Sequencer Setup (SEQ) File. In other words, if no path is specified in the SEQ File, the Sequencer will search the location specified by the SEQINPFILS setting in your CSLSTAR.GLB File for the input files specified in the SEQ File.

In batch mode, the Sequencer can accept a full path and filename to the SEQ File, so the Sequence files do not have to reside in the location specified by the SEQUENCERFILES setting in your CSLSTAR.GLB File. **Important:** When executing the job through the Graphical User Interface, however, it is necessary for the SEQ files to be in the location specified by the SEQUENCERFILES setting (as described below).

As part of the Oracle Utilities Load Analysis system installation, you or your System Administrator must establish two directories for Sequencer-related files. The first directory holds your SEQ files; the second holds all input files specified in those SEQ files. The directories can have any name, and be located anywhere on a local or network drive accessible by both the client and the server. These folders should already be shared over the network allowing access from the client and server. Each client should have read/write access to these two folders.

So that the Sequencer can find the directories you've created for this purpose, the CSLSTAR.GLB Configuration File must contain the following two lines:

```
SEQUENCERFILES <full-path-name>
```

where:

- <full-path-name> is the path to the directory that holds the SEQ files; for example SEQUENCERFILES c:\CSLSTAR\SETUPFLS\.

```
SEQINPFILS <full-path-name>
```

where:

- <full-path-name> is the path to the directory that holds the input files; for example SEQINPFILS c:\CSLSTAR\INPTFLS\.

> **Note**: Path names should be written as the Server machine sees them, not as the Client machine sees them. In addition, the backslash (\) at the end of each line is ***required***.

How to Modify CSLSTAR.GLB:

1. Select **File->PC Server Configuration->Run Time**. The **Server Configuration** window opens.
2. If an entry for SEQUENCERFILES already exists, highlight the line. (If not, type `SEQUENCERFILES` in the **Keyword** box.)
3. In the **Value** text box, type the full path to the directory set up to hold the SEQ files.
4. Click **Add** or **Replace**. (Use Replace if you already have this setting and you want to modify it.)
5. Update or add the SEQINPFILS line with the path to the directory for the input files, using the same method.
6. Click **Save** to save your changes to the CSLSTAR.GLB File and close the dialog box.

   Note: If you later change the name or location of either directory, you must update the CSLSTAR.GLB File accordingly.

Copying Files to the Sequencer Input Folder

Input files for the sequencer can be copied to the input folder by right clicking on an existing file in the Job Results panel and selecting "Copy to Sequencer Input Directory".

In order to enable this functionality you must include the following setting in your .INI file:

```
EnableExploreSeqrInps = 1
```

You must then specify the sequencer input files path under **Tools -> Options.**

This functionality also allows you to open the sequencer input folder in Windows Explorer from the File Composer window or by right-clicking on a sequencer file in the Submit panel.

The SEQ File

This section describes how to create the SEQ file, including the use of variables and conditional statements. This section includes:

- **Creating the SEQ File**
- **Sample SEQ File**
- **Sequencer Variables**

Creating the SEQ File

The first step in taking advantage of the Program Sequencer is to create your SEQ File. The SEQ is a simple text file made up of a series of commands that tell the Sequencer what programs you want to run, where to find and/or how to process the inputs required for each program, and where to put the output.

You can create the file using Notepad or another text editor, or you can create it using the GUI. You must store the file in the directory that's identified in the SEQUENCERFILES Command in CSLSTAR.GLB (as explained on the previous page), and the filename extension must be ".seq".

The following pages take you step-by-step through the process of creating a SEQ File. A completed sample file is shown under **Sample SEQ File** on page 6-14. These instructions assume that you have Notepad or another editor open.

Important: When specifying path names and/or file names that contain spaces, you must enclose the entire name in quotes (" "). For example,

```
OUTDIR "C:\CSLSTAR\OUTPUT AREA"
```

or

```
CTL "C:\CSLSTAR\DATA\CONTROL FILE.ctl"
```

Generating Sequencer Step Commands from Job Results

The SEQ file includes step commands that describe the programs to run and their inputs. You can generate sequencer step commands from a completed job by right clicking in the Job Results panel and selecting "Generate Sequence Step Command" from the context menu.

In order to enable this functionality you must include the following setting in your .INI file:

```
EnableGenerateSequencerSteps = 1
```

The generated commands are based on the job file in the directory and are intended to be used as examples of commands for the SEQ file. You must still supply any necessary path names and place the actual files (such as CTL or ENV files) in the proper sequencer input folders.

How to Create a SEQ File

1. Specify a directory to hold the output for sequences that you will submit in batch mode. The directory must already exist, unless you include the NoDateStamp parameter as described below. If you run this job using the GUI, this line is optional.

   Important! If used, this record must be the first record in the SEQ File.

   The first line in your SEQ File must specify the "Output Directory", which is the parent directory that will hold the subdirectories containing the job results.

   The format of this command is:

   ```
   OUTDIR <directory-name> [NoDateStamp]
   ```

   where:

- **OUTDIR** is a required keyword. This is required when submitting batch SEQ files. Comment out this line when you are running from the GUI and you want the output to reside in the job folder.
- **<directory-name>** is the full pathname to a directory. The directory must already exist, unless you include the NoDateStamp parameter as described below. Once the output directory is specified, you can reference it in a later Step Definition record using the '%JOBDIR%' variable.
- **NoDateStamp** is an optional parameter that may be entered as either NoDateStamp or NDS. Specifying this parameter causes the Sequencer to create <directory-name> and place the results in it. NoDateStamp is case-*in*sensitive, and must be preceded by a space.
  - If you do not specify this parameter the default Date Stamp feature is used. In this case, the <directory-name> you supply must refer to a directory that ***already exists***.
  - If the NoDateStamp option is specified, the <directory-name> you supply must refer to a directory that ***does not exist***. If you supply the name of an existing directory with the NDS option, the job will end with a return code of 115.

Examples:

```
OUTDIR C:\CSLSTAR\SEQNCR

OUTDIR C:\CSLSTAR\SEQNCRD NDS

OUTDIR C:\CSLSTAR\SEQNCRD NoDateStamp
```

At the conclusion of a job run in batch mode, the Sequencer puts the results of the programs in a subdirectory under the directory you specify here. The Sequencer names the subdirectory using the date and time of the run (according to the computer's clock) in the mm.dd_hh.mm.ss format.

If you submit the sequence via the Oracle Utilities Load Analysis Graphical User Interface and supply an Output Directory Record, the results are output to the directory that you defined in the Output Directory Record as described above.

Whether or not you supply an Output Directory Record, when you submit the sequence via the Oracle Utilities Load Analysis GUI, the results are automatically placed in the job directory that you specify when you submit the job. Under that directory, the Sequencer creates a subdirectory for each job step, using the naming convention **STEP<step-number>**.

> **Note:** You can append a meaningful term to the subdirectory name by including a text string after the pound sign (#) at the beginning of the step definition record. See step 6-a below.

2. Next, if running input programs or validation, specify the name and location of the default Environment files.

   Several programs require special default Environment files. These files are typically protected from accidental or casual modification, since they are used by multiple programs and are typically set up according to the policies of your facility (see the Oracle Utilities Load Analysis Load Data Management User's Guide for details).

   These Environment files are:

File	Used by procedures...
Validation Environment File (tgd21b.env)	X110, X310
Reporter Environment File (tgd41b.env)	X110, X210, X220, X310, X410, Y410

File	Used by procedures...
Automatic Editor Env File (tgd31b.env)	X110, X210

If you include any of the procedures listed above in your sequence, you ***must*** specify the required default Environment files in the Default Environment File Records, using the following format:

```
ValidationEnvFile <full-path-and-filename>

ReporterEnvFile <full-path-and-filename>

AutoEditEnvFile <full-path-and-filename>
```

Examples:

```
VALIDATIONENVFILE D:\CSLSTAR\COMMON\DATA\TGD21B.ENV

REPORTERENVFILE D:\CSLSTAR\COMMON\DATA\TGD41B.ENV

AUTOEDITENVFILE D:\CSLSTAR\COMMON\DATA\TGD31B.ENV
```

The Default Environment File Records must follow the Output Directory (if it is supplied) and precede the Step Definition Records.

3. *Optional*. Specify the COMBINE_INP_REPORTS command to control the behavior of the input procedure directing it to either combine or separate the individual reports. Valid values for this command include "YES" (default) and "NO."

4. *Optional* Next, specify parameters for access to interval data stored in Relational tables in the Oracle Utilities Data Repository.

   These parameters are set with the following keywords:

   - RDBUSERNAME: User ID used to connect to the Oracle Utilities Data Repository.
   - RDBPASSWORD: Database password used to connect to the Oracle Utilities Data Repository.
   - RDBCONNECTSTRING: The Data Source Name for the Oracle Utilities Data Repository.
   - RDBQUALIFIER: Database qualifier used to connect to the Oracle Utilities Data Repository.
   - DBPARMSFILE: As an alternative to providing database login information and password in plain text (by using RDBUSERNAME, RDBPASSWORD, etc.), you can use an encrypted database parameter file (DBPARMS.INF) instead. To do this, specify a path to a file name containing the encrypted database connection parameters to connect to the RDB database. If only a filename is provided, then the sequencer will default to the Sequencer Input Files path location. The encrypted database parameters file (DBPARMS.INF) is created and can be found in the job folder whenever a job is submitted from the Load Analysis Client. If DBPARMSFILE is used, all other database connection parameters are ignored.

     Example: DBPARMSFILE "D:\LODESTAR\SEQUENCER\DBPARMS.INF"

   Unless otherwise specified, these values are used for all programs defined in the sequence file. These parameters can also be included in Step Definition records if individual programs need access to different relational data sources.

5. *Optional*. Next, specify which locale setting the sequencer file will use with the following parameter:

   - **LOCALE.** (optional) Use this setting to specify the locale to use. Locale determines date and number formats to use and print. If this setting is not specified and the sequencer is

submitted via the GUI, then sequencer will use the current locale found on the GUI. If not specified and the sequencer is submitted via batch command line, then the sequencer will use the current locale found on the server.

Example:

```
LOCALE [3-Digit-Language-Code]
```

See **Language Codes for LOCALE Setting** on page D-27 to see a list of valid language codes.

6. For each program in the sequence, create a set of commands that defines the program to be run, along with its input files and databases.

   Each set of commands is called a "Step Definition Record" because it defines one step (program) in the "job stream." You must supply the Step Definition Records in the order in which the programs will process. Do not supply step numbers; the Sequencer adds them automatically.

   The following explains the parts of a Step Definition Record, in the order in which you should enter them.

   a. To signal the beginning of a Step Definition Record, enter a pound sign (#). Go to the next line.

      Note:
      You can customize the name of the step's output folder by adding a space and then a string value after the #. For example, if you begin a Step Definition Record with:

      ```
      # MYSTEPNAME
      ```

      The Step Definition Record will create a folder in the output directory with the name "STEPnn-MYSTEPNAME", where "nn" is the step number. The folder name must conform to the Windows folder naming standards. The total length of the folder name is limited to 64 characters.

   b. Specify the input files for the program in this step using one or more "Input-File commands." You can have many Input-File commands in one Step Definition Record, each command specifying one input file. Because there are many types of input files and ways of obtaining and/or pre-processing them, there are several components that you can use to build an Input-File Command. Just the first two (keyword and filename) are required:

      Note: Each File Command must be on a separate line within the step definition.

      ```
      <keyword> <filename> <FROMSTEP n> <OPTxSTEP> <RUN pgm> <PARMS parm 1-3>
      ```

      The first component is a **keyword** that identifies the type of file to be input to the program. For this keyword, you ***must*** use one of the following (case-insensitive):

Keyword	Description
CTL	(.CTL, .INP, .LSE, .KYS) Input file containing data to be processed (required). The file referred to in a CTL Command may be: a Control File, a Load Input File, or a file that was output from an earlier step, such as a Keylist File.
ENV	(.ENV) Environment File.
BNV	(.ENV) Base Environment File.
HOL	(.HOL) Holiday File.

Keyword	Description
TOU	(.TOU) Time Of Use File.
SEA	(.SEA) Season File.
USD	(.USD) User-Specified Day File.
KYS	(.KYS) Input Keys File (Key Generator).
PEA	(.PEA) Peak Day File.
GUF	(.GUF) Generating Units File (Cost of Service Interface - COSI).
DBL	(.DBL) Record Definition File.
CYC	(.CYC) Billing Cycle File.
DEM	(.DEM) Demand File.
CRF	(.CRF) COSI Rates File (must be specified in the SEQ File when required by the program; i.e., there is no default).
CTF	(.CTF) COSI Titles File (must be specified in the SEQ File when required by the program; i.e., there is no default).
RUL	(.RUL) Rules File for X210 (must be specified in order to trigger Auto Editor commands).
DEF	(.DEF) Record Definition file for B210
PDF	(.PDF) Population Data File for B210

Note about Holiday, TOU, Season, Demand, Billing Cycle, and User-Specified Day files: Typically, default versions of these files are specified in the CSLSTAR.GLB Configuration File at system setup. If so, you do not have to specify these files in the Step Definition Records (if none are specified, the Sequencer will look for the defaults in the Step Definition Record).

Note on Required Filename Extensions: During validation of the SEQ file, the sequencer program checks to ensure that filename extensions conform to Oracle Utilities Load Analysis standards and requirements. The required extensions are in parentheses.

The next item in the Input-File Command is the **filename** (also required). The filename identifies the specific file to be input. If you do not provide a full path and filename, one of the following must be true: a) the file must reside in the directory defined by SEQINPFILS in the CSLSTAR.GLB, or b) you can use the following parameters in the Input-File Command to specify that the file comes from an earlier step in the sequence (see below).

If the input file is to come from an earlier step, use the following parameters:

FROMSTEP n. *Optional.* This parameter tells the Sequencer to obtain the file defined by the Keyword from a file produced by Step *n*. For example, say you want to use a Keylist File produced by the second program in the sequence as the Control File for the third program. In that case, you would include a command in the third Step Definition Record similar to the following:

```
CTL TGX31AE.DAT FROMSTEP 2
```

If you provided a customized folder name at the beginning of the step definition record, you can use that name in place of the "n" in the "FROMSTEP n" parameter. For example, if you began an earlier step with "# MYSTEPNAME", you can then use the following command:

```
CTL TGX31AE.DAT FROMSTEP MYSTEPNAME
```

OPT. *Optional.* This parameter tells the Sequencer not to check for the existence of the Input File during the syntax check process. You can use this parameter when the file may not be present until execution of the Sequencer (for example, when the previous step copies the file to a specified area).

OPTxSTEP. *Optional.* If a Control File or Environment File is optional for the program specified in the step, you can use this parameter to tell the Sequencer whether or not to run the step if the specified file does not exist at the time of execution. Replace the *x* value of the parameter with either RUN or SKIP, as described below.

Note: This parameter, if used, must directly follow the FROMSTEP option. If FROMSTEP is not specified then the OPTxSTEP must follow the <filename>. This parameter will be ignored if it is not placed as previously specified, or if it is specified for something other than the CTL and ENV keywords.

- **OPTRUNSTEP.** This parameter tells the Sequencer to run the step even if the file specified does not physically exist when this step is executed.
- **OPTSKIPSTEP.** This parameter tells the Sequencer to skip the step if the specified file does not exist at time of step execution. This option will also place the following message in the REPORT.HTML File in the directory for the step: "This step was skipped because the OPTSKIPSTEP option was specified".

If the file identified in the Input-File Command requires pre-processing before it is input to the step program (such as to update the date value in the Environment File), use the RUN and PARMS parameters to specify the pre-processing as described below.

- **RUN <pgm>.** This parameter tells the Sequencer to pre-process an input file by executing the program <pgm>. For example, you might want to use the CSMODENV Program to modify the date parameters in an Environment File before using that Environment File as input to the program. (See **Specifying Dates in the Sequencer** on page 6-26 for a description of two of the programs you can use to pre-process input files in this way.)

  The RUN command can only be used with CTL, ENV and BNV command lines. If the RUN command is followed by a custom executable, the executable file must either be located in the BINS folder on the server or you must include a fully qualified path.

- **PARMS <filename> <SUPPDATE.DAT | FROMSTEPn>.** These parameters tell the program called by RUN what file to pre-process. This file will typically be either the SUPPDATE.DAT File described in the next section, or a file output by a previous step. ***Note:*** If the RUN keyword is not specified, then PARMS keyword and any values following it will be ignored. Similarly, if the PARMS keyword is omitted, any values after the <pgm> value of the RUN keyword will be ignored. When using PARMS, the filenames must be fully qualified. The filename must also be the first parameter listed if there are more than two or more parameters.

  For example, the following command would modify the DAT Command in TGD72B.ENV with the date found in a file that was output by Step 2 — TGX671.DAT.

```
ENV TGD72B.ENV RUN CSMODENV PARMS TGX671.DAT FROMSTEP 2
```

c. Once you have specified all of the necessary Input-File commands for the Step Definition, specify the database(s) containing the data to be processed in the step and (depending on the program) the database to write the results to.

Use a Database Command to specify each database involved in the step.

The format for the database command is as follows:

```
DBA <database-table-name> <databasetype>
```

where:

DBA is the required keyword for the DBA Command.

database-table-name is the name of the database table in the Oracle Utilities Data Repository. Note that spaces cannot be used in the table name here.

Note: Previous versions required that database table names be prefixed with "RBD/". This is no longer required, but is still supported.

databasetype is any one of the database types recognized by Oracle Utilities Load Analysis, including **CLDB** (for Edit Database), **AXDB** (for Auxiliary Database), **ELDB** (for Extract Database), **GLDB** (for Settlement Database), and **CLDB2** (for temperature database), as well as SLDB, RLDB, and ALDB. You can also specify the database as **SOURCE** or **TARGET.** For programs that can read from multiple databases, you can also include **SOURCE2** and **SOURCE3**

For example:

```
DBA LSCHCLDB SOURCE
```

d. In the Step Definition Record, you *must* specify the program or procedure to run in the step. You do that using a **Program Command**. A Step Definition Record can contain just one Program Command.

There are two formats for Program commands, depending upon whether the program or procedure is Oracle Utilities or non-Oracle Utilities (i.e., user-written).

Format for Program Commands (Oracle Utilities Load Analysis):

```
APP <applid>
```

Where:

APP is the required keyword for the command.

applid is the ID of the job (see **Appendix D**: **Oracle Utilities Load Analysis Sequencer Program Input Tables and Keywords** or the *Oracle Utilities Load Analysis Quick Reference Guide*, for a list of IDs).

For example, to specify Direct Input:

APP X110

Using the Repeater with the Sequencer

To run Repeater jobs (see **Chapter 7**: **The Repeater**), use the following syntax:

```
APP REPEATER <applid>
```

Where:

APP is the required keyword for the command.

applid is the ID of the job (see **Appendix D**: **Oracle Utilities Load Analysis Sequencer Program Input Tables and Keywords**, or the *Oracle Utilities Load Analysis Quick Reference Guide*, for a list of IDs).

Format for Program Commands (other than Oracle Utilities Load Analysis)

```
PGM <full-pathname> <parameters>
```

where:

PGM is the required keyword for the command.

full-pathname is the name and location of the executable program.

parameters are any parameters required by the program. To reference the output directory (OUTDIR), you can use the '%JOBDIR%' variable.

```
PPM <full-pathname> <parameters>
```

where:

PPM is the required keyword for the command.

full-pathname is the name and fully qualified location of the parameters file. This value can also be the name of a parameters file in the SEQINPS folder.

parameters are any parameters required by the program. To reference the output directory (OUTDIR), you can use the '%JOBDIR%' variable.

Note: The parameters file used by the PPM command will override the parameters in the PGM command.

e. Specify a file name for the output file using the **OUT1** command. The format for this command is:

```
OUT1 <filename>
```

where:

OUT1 is the keyword for the command

filename is the name of the output file. Note that you cannot specify a path using this command.

This command controls the name of the output file for the following programs

Program	Output File affected
Key Generator	Output Keys File
Copy Cuts	Keys List file
Prod / Direct Input	Keys List file (TGX112.DAT)
Validation	Edit Keys File
Invalid Series Validation	Editor Keys List
Editor Syntax Scan	Valid requests (TGX322.DAT)
Billed Energy	Analysis Control File (TGY231)
Cust Data Extraction	Keys List File (TGY247)
MPU Analysis	ICS File (TGY318)
Ratio Analysis	ICS File (TGY338)
Domains MPU Analysis	ICS File (TGY358)
Domains Ratio Analysis	ICS File (TGY368)
Individual Cust Analysis	DLM File (TGY377)

Program	Output File affected
100% Sample Analysis	ICS File (TGY388)
Direct Output	Extracted Data File

f. If desired, you can include a Return Code Command in the Step Definition that specifies a threshold for aborting the job stream.

The format is:

```
STP <n>
```

where:

STP is the required keyword for the command (enter as shown).

n is the maximum allowable return code. If the return code from the program equals or exceeds **n**, the job will abort at that step. The default is 99. A list of return codes and their meanings is provided in the last section of this chapter.

g. You have completed your first Step Definition Record. Repeat this process for every program in the sequence (for a total of up to 99 programs).

7. You may include comments anywhere in the file, as a helpful reminder to yourself or others who might use or review your SEQ File. You can even add a comment on the same line as a command. (See **Sample SEQ File** on page 6-14.)

Comments are ignored by the programs. A comment must begin with a slash, asterisk, and be followed by at least one blank space (/*).

You may have as many comments as you wish.

8. Save your file. Remember, if submitting the job through the GUI, you must place the file in the directory that's identified in the SEQUENCERFILES Keyword in CSLSTAR.GLB, and the file must have the '.seq' extension.

Sample SEQ File

```
OUTDIR C:\CSLSTAR\SEQNCR
VALIDATIONENVFILE D:\CSLSTR\COMMON\DATA\TGD21B.ENV
REPORTERENVFILE D:\CSLSTR\COMMON\DATA\TGD41B.ENV
AUTOEDITENVFILE D:\CSLSTR\COMMON\DATA\TGD31B.ENV

# /* STEP 1: DIRECT INPUT OF DAILY SYSTEM LOAD AND MONITORED LOADS
CTL 120182.INP
ENV TGD11B.ENV
RUL TGD21C.RUL RUN FILEPREP
DBA LPSSCLDB CLDB
DBA TSTAXDB AXDB
APP D110

# /* STEP 2: VALIDATION
CTL TGX112.DAT FROMSTEP 1
ENV LPSX21B.ENV
RUL TGX21C.RUL RUN FILEPREP
DBA TSTAXDB AXDB
DBA LPSSCLDB CLDB
APP X210

# /* STEP 3: AUTO EDITOR
CTL TGX31AE.CTL FROMSTEP 2  OPTSKIPSTEP
ENV LPSX31B.ENV
DBA TSTAXDB AXDB
DBA LPSSCLDB CLDB
APP X310

# /* STEP 4: DATA EXTRACTION TO PERFORM STATISTICAL EXPANSION OF SAMPLE POINTS
CTL Y240LPSS.CTL
ENV TGY24B.ENV
DBA LPSSCLDB CLDB
DBA LPSSELDB ELDB
APP Y240

# /* STEP 5: STANDARD LOAD ANALYSIS RATE 1  GENERATING CLASS TEMPLATES
CTL Y31ALPSS.CTL
ENV Y31BLPSS.ENV
DBA LPSSELDB CLDB
DBA LPSSELDB CLDB2
APP Y310

# /* STEP 6: STANDARD LOAD ANALYSIS RATE 2  GENERATING CLASS TEMPLATES
CTL Y31LPSS2.CTL
ENV Y31LPS2.ENV
DBA LPSSELDB CLDB
DBA LPSSELDB CLDB2
APP Y310

# /* STEP 7: ELDB TO GLDB EXTRACTION OF DATA FROM TEMPLATES
CTL Z120LPSS.CTL
ENV Z120LPSS.ENV
DBA LPSSELDB CLDB
DBA LPSSGLDB GLDB
APP Z120

# /* STEP 8: PROXY DAY
ENV PROXYSEQ.ENV
SEA TGE31E.SEA
HOL TGE31C.HOL
DBA TSTAXDB AXDB
DBA LPSSELDB ELDB
APP X670
```

1
2
3
4
5

Sample SEQ File Legend

1. Output Directory Command: Tells the Sequencer where to put the job results. This is required if you submit the program sequence in the batch mode, but optional if you submit it via the GUI.

   The directory specified must already exist. Oracle Utilities Load Analysis will not create it automatically.

2. Default Environment File commands: Tell the Sequencer where to find the default Environment files required for some steps. These commands are *required.*

3. A pound sign (#) indicates the beginning of a Step Definition Record.
4. Step Definition Records: Specify the procedure or program to run, and any user-specified inputs that are required for that procedure or program. SEQ files can contain up to 99 steps.
5. The Slash-Asterisk indicates the beginning of a Comment. Anything following the /* will be ignored by the program.

   If you would like to number your sequence steps you may do so automatically by pressing together the CTL-ALT-3 keys while in the sequencer editor screen. By doing this you may easily identify your steps. The function will insert a "STEP nn" comment for each Step identifier (#) in your sequence.

Sequencer Variables

This section describes how to use the sequencer with variables, including:

- **Overview of Sequencer Variables**
- **Using Sequencer Variables**
- **Sequencer Functions**
- **Passing Parameters to the Sequencer**

Overview of Sequencer Variables

The **Sample SEQ File** on page 6-14 uses hard-coded values for the sequencer commands. An alternative is to use sequencer variables, which allow you to dynamically substitute values in the file before it is executed. The use of variables also allow a single SEQ file to be used for multiple purposes. The sequencer variables are defined in the header section of the SEQ file.

There are two types of sequencer variables:

- **Local:** A local variable is a variable whose scope is within the sequencer file and the input files, such as control and environment files. Local variables are declared with a **VAR** statement.
- **Parameter:** A parameter variable is the same as a local variable except its value can be passed as a parameter in a command line argument, by using a configuration file, or when submitting from the GUI Job Parameters Input screen that is displayed to the user when the sequencer is run. See **Passing Parameters to the Sequencer** on page 6-24 for more information. Parameter variables are declared with a **PAR** statement.

The formats for these commands are:

```
PARameter variable-name = <default-value> [; description]
VARiable variable-name = <value> [; description]
```

Where:

variable-name specifies the name of the variable. The name can contain only letters, numbers, or underscores, and cannot be longer than 32 characters. Spaces or other special characters are not allowed.

value specifies the default value to be used for the variable. Values should be enclosed in double quotes (“value”). The value initially assigned to the variable remains until the value is changed by way of a command line argument, when a user changes it when submitting from the GUI, or by using a SET statement later in the SEQ file.

description (optional) specifies a short description for the defined variable. The description will be displayed in place of variable name when the sequencer is submitted using the GUI in the Job Parameters window. The description should be enclosed in double quotes (“description”). If a description is specified, a semi-color delimiter (;) must be used to delimit it from the default-value. If a description is omitted, the GUI displays the variable-name to the user.

Examples:

```
PAR START_TIME = "01/01/12"; "EXTRACT RANGE"
PAR STOP_TIME = "01/31/12"
VAR RATE_CLASS = "RES"
```

Using Sequencer Variables

Once sequencer variables are declared they are available for use in all subsequent sequencer input files (for example, control and environment files) and within the sequencer file itself. These variables are referenced by surrounding the variable-name with percent symbols (%).

For example, your sequencer file may have the following:

```
PAR START_TIME = "01/01/12"; "EXTRACT RANGE"
PAR STOP_TIME = "01/31/12"
VAR DATABASE = "CLDB"

# /* DIRECT OUTPUT
CTL X720.CTL
ENV X720.ENV
DBA LSCH%DATABASE% CLDB
APP X720
```

The sequencer variable DATABASE is declared with a value of "CLDB". %DATABASE% is a reference to the sequencer variable which will be substituted with a value of the variable, "CLDB", during the execution of the file. Because the LSCH is required for all interval tablesets, we can simply use variable for the unique part of the string.

The sequencer input environment file, X720.ENV, can have the following:

```
AGG 3600
DATE %START_TIME% %STOP_TIME%
```

In this example, the references to sequencer variables %START_TIME% and %STOP_TIME%, which were defined in the sequencer file, will be replaced with their respective values. In this example, "01/01/12 01/31/12" will be used in the DATE command.

Changing the Value of a Variable with the SET Command

You can change the values of sequencer variables between steps by using the **SET** command. This allows you to use one variable multiple times but with different values.

The SET command has the following format:

```
SET <variable-name> = <new-value>
```

Where:

variable-name specifies the name of the variable that holds the value you want to change.

new-value specifies the new value to assign to the sequencer variable. If the new value contains spaces it must be contained in double quotes.

Example:

```
SET RATE_CLASS = "COMM"
```

Example SEQ file Using Variables

```
OUTDIR C:\LODESTAR\LODESTAR111\SEQOUT\

PAR START_TIME = "01/01/12"
PAR STOP_TIME  = "01/31/12"
VAR RATE_CLASS = "RESD"

# /* STEP 1
CTL Y330A.CTL
ENV Y330.ENV
DBA LSCHELDB ELDB
APP Y330

# /* STEP 2
```

```
SET RATE_CLASS = "COMM"
CTL Y330B.CTL
ENV Y330.ENV
DBA LSCHELDB ELDB
APP Y330
```

The sequencer input environment file, Y330.ENV, for the sequencer above can contain the following:

```
ALPHA 10%
ASSIGN FIXED
DATE %START_TIME% %STOP_TIME%
KEY %RATE_CLASS%
STRATA 1 4254321 514.0 8931
STRATA 2 6569052 919.0 9779
STRATA 3 6679929 INF 4638
```

In this example, %RATE_CLASS% referenced in the Y330.ENV file will have a value of "RESD" for Step 1 and a value of "COMM" in the subsequent Step 2. Note that the DATE command will be use "01/01/12 01/31/12" for both steps 1 and 2.

Using Conditional Statements

You can perform conditional processing with sequencer variables by using the **IF** statement. The IF statement has the following format:

```
IF sequencer-variable <operator> value THEN command
```

Where:

Sequencer-variable refers to an existing sequencer variable to evaluate. Specify the variable name you wish to compare. The variable name should not be enclosed in percent symbols (%).

<operator> specifies a comparison operator to use in the evaluation. Available operators include:

Operator	Evaluation
=	Returns true if the sequencer variable is equal to the specified value.
>	Returns true if the sequencer variable is greater than the specified value.
<	Returns true if the sequencer variable is lesser than the specified value.
>=	Returns true if the sequencer variable is greater than or equal to the specified value.
<=	Returns true if the sequencer variable is less than or equal to the specified value.
!=	Returns true if the sequencer variable is not equal to the specified value.

value specifies the value to which the sequencer variable is to be compared. String values should be enclosed in single or double quotes.

command specifies a sequencer command statement to be executed if the evaluation returns true.

Example:

```
IF BILLPERIOD = "1" THEN SET CYCLEFILE = "JAN.CYC"
```

Example Sequencer File Example Using Conditional Statements:

```
PAR BILLPERIOD = "1";"BILL PERIOD"
VAR CYCLEFILE = "Y230.CYC"
VAR CTLFILE = "Y230.CTL"

IF BILLPERIOD = 1 THEN SET CYCLEFILE = "JAN.CYC"
IF BILLPERIOD = 2 THEN SET CYCLEFILE = "FEB.CYC"
IF BILLPERIOD = 3 THEN SET CYCLEFILE = "MAR.CYC"
IF BILLPERIOD = 4 THEN SET CYCLEFILE = "APR.CYC"
IF BILLPERIOD = 5 THEN SET CYCLEFILE = "MAY.CYC"
IF BILLPERIOD = 6 THEN SET CYCLEFILE = "JUN.CYC"
IF BILLPERIOD = 7 THEN SET CYCLEFILE = "JUL.CYC"
IF BILLPERIOD = 8 THEN SET CYCLEFILE = "AUG.CYC"
IF BILLPERIOD = 9 THEN SET CYCLEFILE = "SEP.CYC"
IF BILLPERIOD = 10 THEN SET CYCLEFILE = "OCT.CYC"
IF BILLPERIOD = 11 THEN SET CYCLEFILE = "NOV.CYC"
IF BILLPERIOD = 12 THEN SET CYCLEFILE = "DEC.CYC"

# /* BILLED ENERGY PROGRAM
CTL %CTLFILE%
ENV Y230.ENV
CYC %CYCLEFILE%
DBA LSCHELDB SOURCE
APP Y230
```

Sequencer Functions

Sequencer functions are functions that can be used within a SET command to manipulate strings or sequencer variables. All sequencer functions return values in the form of a string, which in turn is assigned to an existing sequencer variable by using the SET command. Sequencer functions have the following format:

```
SET variable = function()
```

The following table describes the sequencer functions:

Function	Description

ADD_DAYS()	Adds a number of days to a supplied date string and returns the result in the form of a date string ("MM/DD/YY" or "MM/DD/YY-HH:MM:SS"). This function has the following format:
	`ADD_DAYS(date; n)`
	Where:
	• **date** specifies the date to be operated on. Must conform to MM/DD/YY or MM/DD/YY-HH:MM:SS format, with respect to locale.
	• **n** specifies the number of months/days/years to add to date. For subtraction operation, indicate a negative value.
	Examples:
	`SET STARTDAT = ADD_DAYS("12/01/13"; 1)`
	The above function returns "12/02/13"

Function	Description

ADD_MONTHS()	Adds a number of months to a supplied date string and returns the result in the form of a date string ("MM/DD/YY" or "MM/DD/YY-HH:MM:SS"). This function has the following format:
	`ADD_MONTHS(date; n)`
	Where:
	• **date** specifies the date to be operated on. This value must conform to MM/DD/YY or MM/DD/YY-HH:MM:SS format, with respect to locale.
	• **n** specifies the number of months/days/years to add to date. For subtraction operation, indicate a negative value.
	Examples:
	`SET STARTDAT = ADD_MONTHS("12/01/13"; 1)`
	The above function returns "01/01/14"
	`VAR JAN = "01/01/14"`
	`SET STARTDAT = ADD_MONTHS(%JAN%; 1)`
	The above function returns "02/01/14"

ADD_YEARS()	Adds a number of years to a supplied date string and returns the result in the form of a date string ("MM/DD/YY" or "MM/DD/YY-HH:MM:SS"). This function has the following format:
	`ADD_YEARS(date; n)`
	Where:
	• **date** specifies the date to be operated on. Must conform to MM/DD/YY or MM/DD/YY-HH:MM:SS format, with respect to locale.
	• **n** specifies the number of months/days/years to add to date. For subtraction operation, indicate a negative value.
	Examples:
	`SET STARTDAT = ADD_YEARS("12/01/13"; 1)`
	The above function returns "02/01/14"

Function	Description

BEGIN_MONTH ()	Given a supplied date string, returns a date representing the first day of month. Note this function converts a given date-time to the first day of the month only. The time, if provided, is not adjusted in any way. For example, if a time of 8am is provided, the returned date will be the first day of the month, at 8am.
	
	The result is returned in the form of a date string ("MM/DD/YY" or "MM/DD/YY-HH:MM:SS"). This function has the following format:
	
	`BEGIN_MONTH(date)`
	
	Where:
	• **date** specifies the date to be operated on. Must conform to MM/DD/YY or MM/DD/YY-HH:MM:SS format, with respect to locale.
	
	Examples:
	`SET STARTDAT = BEGIN_MONTH("12/13/13")`
	
	The above function returns "12/01/13"
	
	`SET STARTDAT = BEGIN_MONTH("12/21/13-08:00:00")`
	
	The above function returns "12/01/13-08:00:00"

END_MONTH ()	Given a supplied date string, returns a date representing the last day of month. Note this function converts a given date-time to the last day of the month only. The time, if provided, is not adjusted in any way. For example, if a time of 8am is provided, the returned date will be the last day of the month, at 8am.
	
	The result is returned in the form of a date string ("MM/DD/YY" or "MM/DD/YY-HH:MM:SS"). This function has the following format:
	
	`END_MONTH(date)`
	
	Where:
	• **date** specifies the date to be operated on. Must conform to MM/DD/YY or MM/DD/YY-HH:MM:SS format, with respect to locale.
	
	Examples:
	`SET DATE = END_MONTH("12/13/13")`
	
	The above function returns "12/31/13"
	
	`SET DATE = END_MONTH("12/21/13-08:00:00")`
	
	The above function returns "12/31/13-08:00:00"

Function	Description

LOOKUP()	Performs a lookup operation on a supplied XML file. The resulting value is returned as a string. This function has the following format:
	`LOOKUP(value, table-name, xml-filename)`
	Where:
	• **value** specifies the value that LOOKUP searches for in the table (Required)
	• **table-name** specifies the name of the table from which to perform the search in (Required).
	• **xml-filename** specifies the name of the Lookup XML file to use for this lookup. Unless a full path is specified, LOOKUP will default to your Sequencer Input folder location (SEQINP) (Required). See **Lookup XML File Format** on page 6-23 for more information about the format of this file.
	Example:
	`SET START = "01/31/16"`
	`SET MM = SUBSTR(%START%, 1, 2)`
	`SET LABEL = LOOKUP(%MM%, "MONTH", "LOOKUP.XML")`
	The above function returns "JANUARY" (see example XML file below).

MATH()	Performs a math operation on a supplied expression. The resulting value is returned as a string. This function has the following format:
	`MATH(expression; [decimal_digits])`
	Where:
	• **Expression** specifies a mathematical expression to be evaluated and performed operation on.
	• **Decimal_digits** (optional) specifies the number of decimals to format the result to. By default, the number of decimals is 0 (rounded to the nearest whole number).
	Examples:
	`SET RESULT = MATH(1 + 2)`
	The above function returns "3".
	`SET NUM = "5"`
	`SET RESULT = MATH(%NUM%/2)`
	The above function returns "3".
	`SET NUM = "5"`
	`SET RESULT = MATH(%NUM%/2; 2)`
	The above function returns "2.50".

Function	Description

SUBSTR()	Performs a substring operation on a supplied value and returns a subset of the original string. This function has the following format:
	`SUBSTR(text; start_pos; [length])`
	Where:
	• **Text** - specifies the string to be operated upon.
	• **Start_pos** specifies the starting character position for the operation. The first character is position 1.
	• **Length** (optional) specifies the number of characters to be returned. If not provided, all characters from the start position to the end of the string will be returned.
	Examples:
	`SET RATE = SUBSTR("ABCDEFG"; 1; 3)`
	The above function returns "ABC".
	`SET ID = "ABC123"`
	`SET RATE = SUBSTR(%ID%;4)`
	The above function returns "123".
	Predefined Date/Time Variables
	The SUBSTR function can use the following predefined variables:
	• **%TODAY%**: Stores the current date (MM/DD/YY).
	• **%NOW%**: Stores the current date-time (MM/DD/YY-HH:MM:SS).
	The date formats returned are based on the current locale that is set the by the sequencer or the GUI.

Lookup XML File Format

The LOOKUP function references a table configured in an XML file with the following structure. This file should be located in the Sequencer Input file location (defined via the SEQINP GLB setting).

Lookup XML File Example

```
<SEQUENCER>
  <LOOKUP ID="MONTH">
    <KEY ID="01" VALUE="JANUARY"/>
    <KEY ID="02" VALUE="FEBRUARY"/>
    <KEY ID="03" VALUE="MARCH"/>
    <KEY ID="04" VALUE="APRIL"/>
    <KEY ID="05" VALUE="MAY"/>
    <KEY ID="06" VALUE="JUNE"/>
    <KEY ID="07" VALUE="JULY"/>
    <KEY ID="08" VALUE="AUGUST"/>
    <KEY ID="09" VALUE="SEPTEMBER"/>
    <KEY ID="10" VALUE="OCTOBER"/>
    <KEY ID="11" VALUE="NOVEMBER"/>
    <KEY ID="12" VALUE="DECEMBER"/>
  </LOOKUP>
</SEQUENCER>
```

Lookup XML File Element Descriptions

SEQUENCER: The root element of the lookup XML file.

Elements:

- **LOOKUP**: Specifies a lookup table. Each LOOKUP element contains a group of KEYS.

  Attributes:

  - **ID**: Defines a unique table name for this lookup.

  Elements

  - **KEY**: Represents a lookup definition (key-value pair). There can be multiple KEY elements; however the ID for each KEY must be unique to the table.

    Attributes:

    - **ID**: Specifies a unique id for this key.
    - **VALUE**: Specifies the value to be returned for this key.

Passing Parameters to the Sequencer

There are several ways values can be passed into a sequencer job with parameter variables. If the sequencer is submitted as a command line batch job you can use **Command Line Arguments** or a **Configuration File**. If the sequencer is submitted using the Client GUI you can use the **Client GUI Job Parameters Input Screen**.

Command Line Arguments

If your sequencer file contains just a few Parameter Variables, you can submit parameter values using the command line argument.

Format:

```
cslsServer.exe sequencer <SEQ file> [<parameter variable>
<value>] [<parameter variable> <value>]
```

Where:

SEQ file specifies the name of the sequence file (*.SEQ) you wish to run.

parameter variable specifies the parameter variable you want to pass a value to.

value specifies the value you wish to assign to the parameter variable. Enclose value in double quotes (“value”) if the value contains embedded spaces. Include a [<parameter variable> <value>] pair for each parameter variable that you want to pass a value to.

Example:

```
cslsServer sequencer test.seq START_TIME 03/01/13
STOP_TIME 03/31/13
```

Configuration File

If your Sequencer file contains many parameters, you can consolidate all of the parameter values into a .DAT configuration file and submit it as a command line argument.

Format:

```
cslsServer.exe sequencer <SEQ file> [configuration file]
```

Where:

sequence file specifies the name of the Sequence file (*.SEQ) you wish to run.

configuration file specifies the location of the configuration file you wish to use. Enclose in double quotes ("location") if the value contains embedded spaces.

Examples:

```
cslsServer sequencer test.seq d:\cslstar\config.dat
```

```
cslsServer sequencer test.seq "d:\cslstar\sequencer input\config.dat"
```

Configuration File Format:

The configuration file must be saved with a .DAT extension. Each line in the configuration file consists of a parameter name, value pair with the following format:

```
<parameter variable> = <value>
```

Where:

parameter variable specifies the parameter variable you wish to pass a value to.

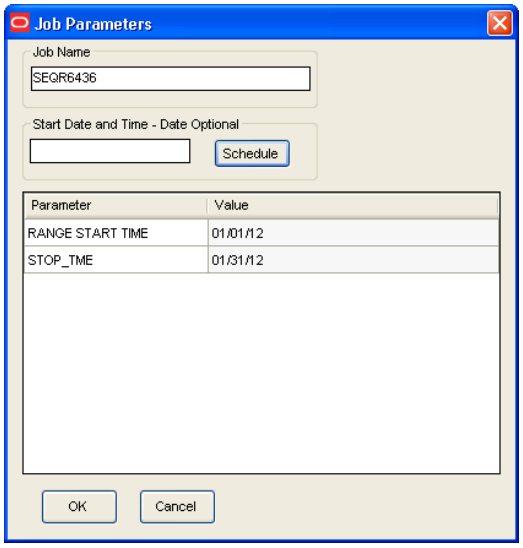
value specifies the value you wish to assign to the parameter variable. Enclose value in double quotes ("value") if the value contains embedded spaces.

Example configuration file (config.dat):

```
START_TIME = 03/01/13
STOP_TIME  = 03/31/13
```

Client GUI Job Parameters Input Screen

When a sequencer job with parameter variables is submitted using the Client GUI, a Job Parameters Input dialog is displayed to allow for parameter value input or override. An entry in the table will be populated for each PARameter variable declared in the SEQ file:



The **Parameter** column displays the name or description of the parameter variable that is defined in the SEQ file. If a description is provided in the PAR statement in the SEQ file, then the description will be displayed. If no description is provided, the name of the variable is displayed.

The **Value** column specifies the value to assign to the parameter. The default value specified in the PAR statement in the SEQ file is displayed. Assign a value of blank ("") in your PAR statement if you do not wish to assign a default value.

Specifying Dates in the Sequencer

Note: Most of this functionality is kept for version compatibility. It is recommended that you use variables whenever possible. For additional information, see **Sequencer Variables** on page 6-16.

Many (if not most) of the Oracle Utilities Load Analysis programs that you're likely to include in a sequence require you to specify a date or date range for the data to be processed. For example, the analysis programs require you to specify the date range of the analysis period; reporting programs require you to specify the dates to be reported. Depending on the program, these dates are typically specified in the Date Command in the program's Environment File. (Or, if you are submitting the job using the Graphical User Interface, you can specify the dates via the Submit Panel.) However, if you are designing the program sequence to run automatically in batch mode on a daily, weekly, or monthly basis, a single static date or date range "hard-coded" in the Environment File would not be adequate.

- Fortunately, there is a more flexible means of specifying dates. Oracle Corporation has provided the CSMODENV utility which is designed specifically for modifying the dates specified in an Environment File's DATe and CDA commands just before the program for the step is run. You can use the Input-File Command's Run parameter (described on page 6-10) to specify that that this utility "pre-processes" an Environment File before it is input to the program for the step. The date applied by these utilities to the Environment File can come from one of several sources:
- An output file from a previous step
- An external file called SUPPDATE.DAT (SUPPDATE.DAT is a simple text file containing two dates). The SUPPDATE file, if used, must be specified after "RUN CSMODENV"

CSMODENV

CSMODENV changes the DAT Command in an Environment File to the date(s) stored in a file that you specify with the Input-File Command's PARMS parameter. The file containing the dates can be output from a previous step; alternatively, you can use the SUPPDATE.DAT File.

Example:

```
ENV TGD72B.ENV RUN CSMODENV PARMS TGX671.DAT FROMSTEP 2
```

The above example modifies the DAT Command in TGD72B.ENV with the date found in a file output by Step 2 — TGX671.DAT. If the Sequencer cannot find the file, it issues a return code 9400 + *n* (where *n* is the step number).

SUPPDATE.DAT

SUPPDATE.DAT is a simple text file containing just a start-date and a stop-date — both in the mm/dd/yy format. For example:

```
01/01/99  01/31/99
```

The purpose of SUPPDATE.DAT is to provide a consistent place where CSMODENV can find the dates to be applied. To change the dates in the file, you can either update them manually using Notepad or another editor, or your System Administrator may set up a routine to update the file automatically. For example, your facility might create a routine that modifies the dates to match the date on the system clock. In either case, SUPPDATE.DAT must reside in the SEQINPFILS directory.

A Note about the date files:

A date file is a simple text file that contains just a start-date and a stop-date. Both dates must be in the mm/dd/yy format. For example:

```
01/01/99 01/31/99
```

Running the Sequencer via Windows Schedule Service

Once you have created the SEQ and input files as described on the previous pages, you can use the AT Command included in the Windows Schedule Service to run the sequence in batch mode. You could schedule it to run once at a future date and time, or repeatedly every time a particular day of the week or date of the month occurs.

Note: Oracle Utilities Load Analysis does not provide security when submitting and/or running a sequence in batch mode. Security and access rights when submitting and/or running a sequence in batch mode is handled by restricting access to the application server through Windows security.

Optionally, you may use the Windows Scheduler GUI on the application server and avoid the complications of using the AT command.

Setting Up the AT Command Batch File

In order to use the AT Command to run a job stream, you must create a batch file containing the following commands:

```
C:

CD C:\CSLSTAR\BINS

CSLSSERVER SEQUENCER %1
```

Modify the first line to point to the drive that contains your BINS directory. Modify the second line to point to your BINS directory. Leave the last line as-is. The first two commands change the current directory to the directory specified (this should be your BINS directory where all of the Oracle Utilities Load Analysis programs and executables reside). The last command triggers the Sequencer Program using the SEQ File you will specify with the AT Command (explained below).

You must store this batch file on the Server machine.

Using the AT Command to Schedule a Program Sequence

Note: The Windows Schedule Service must be running to use the AT Command. See your Windows documentation for instructions.

At a DOS prompt, type the following command to schedule the job:

```
AT  <time> <batchfile> <SEQ-file> <">errorfile">
```

where:

AT is the required keyword for the command.

time specifies the time the job is to run; expressed in hours:minutes. 00:00 represents midnight; 23:59 is the last minute of the day.

You can add either of the following switches to the time parameter to further specify when the job is to be executed:

/next:*date(s)* specifies that the job is to be run once at the indicated time on the next occurrence of each day you provide in a comma-separated list. You can specify days of the week and/or dates of the month. For example,

```
12:00 /next:1,15,Friday
```

means run the job at noon on the next 1st and 15th day of the month (could be this month or next), as well as the next Friday.

/every:*date(s)* specifies that the job is to be run each time the specified days of the week and/or dates in the month occur. For example,

```
12:00 /every:1,15,Friday
```

means run the job at noon on the 1st and 15th day of every month, as well as every Friday.

batchfile is the AT Command batch file described in the section above. Must be fully qualified.

SEQ-file is the name of the Sequence Setup (SEQ) file you wish to run.

"**errorfile**" is the user-specified destination (full path and filename) for program and error messages produced by the job. Must have double-quote marks surrounding it, and begin with the pointed bracket >, as shown below.

Example:

```
AT 14:00 c:\batchfiles\seqsetup seqfile ">c:\cslstar\sqncr\error.log"
```

The example command above schedules SEQSETUP from the batchfiles directory to run at 2:00 pm using SEQFILE.SEQ. In this example, the Sequencer will write any errors or messages encountered in the run to:

```
c:\cslstar\sqncr\error.log
```

The job will run in background mode without additional user intervention, as long as the Windows Schedule Service is properly set up and running. (See your Windows documentation for details.)

Sequencer Security

Improper use of the sequencer can lead to the execution server freezing up, and other performance issues. This section outlines how to set up restrictions to what non-administrator users have access to do through the sequencer, including restricting which programs may be called using the PGM command, and restricting access to the sequencer files.

Sharing Sequencer Files

Preliminary security recommendations include sharing the SEQFILES (the location of the sequencer files (.seq) on the server) and SEQINPS (the location of the sequencer input files on the server) folders on the application server to administrator users with full access. These folders should be shared to non-administrator users using read/only attributes. The server share to the .SEQ files must be named the folder name of the SEQUENCERFILES GLB setting. For example, if your GLB setting for SEQUENCERFILES is d:\lodestar\lodestar111\SEQRFILES, then the server share name must be SEQRFILES so that the clients may automatically connect.

It is the administrators of the system to decide or determine who has what access to the SEQINPS folder, we recommend that administrator have read/write and non-admins have read only or none.

Security Configuration File

Administrator users may optionally create a configuration file called oula_security.cfg.xml in the **LODESTAR111\SERVER\BINS** directory on the server that specifies the allowable executables to be run using the PGM command in the SEQR. See **oula_security.cfg.xml** on page 1-15 in the *Oracle Utilities Load Analysis Configuration Guide* for more information about this file.

Without this file the sequencer will be able to run any program. This file restricts the sequencer to specifically listed programs. The only attributes that may be changed within the SEQR tag are the names of the files allowed. On the server, these files must be fully qualified if the executable does not reside in the server BINS folder. (The sequencer checks for the existence of the file prior to execution and if not fully qualified, it will assume that the file resides in the BINS folder).

The SEQUENCER_SECURITY GLB command controls the security access features. Valid values for this setting are YES and NO (default). When set to YES, this setting application behaves as follows:

- **Server**: Sequencers submitted from batch will conform to the security settings in the oula_security.cfg.xml configuration file if it exists. The Sequencer will check the requested executable on the PGM line to ensure that it is present in the list of approved executables in the configuration file.
- **Client**:
  - If the user is an administrator, all sequencer features will be available. The Administrator may create and edit sequencer files and also change the path to where the gui locates the seq files.
  - If the user is non-administrator (defined in the cslstar.usr file as <userid> ALL:ALL or other non-administrator restrictions), they will have the ability to select, view/preview and submit sequencers already set up in the SEQINPS folder on the server. Users of this type cannot change the SEQ files path. It will be hard coded to \\server-name\SEQRFILES share.

Sequencer Output

The output is placed in the directory that you specified using the Base Output Directory record. Alternatively, if you submitted the job via the GUI and did not provide a Base Output Directory record in the SEQ File, the output is in the job directory that you named using the **Job Parameters** dialog box.

In addition, the program will create a SETUP.REP File in the root job directory, which contains a report of the compilation of the setup file.

Return Codes from Sequencer

The table below lists the return codes returned by the Program Sequencer.

Return Code	Description
0	Successful for all runs.
99	Program terminated abnormally.
101	The setup file for (.SEQ) not present in Sequencer Area.
110	CSLSTAR.GLB File not present in current directory.
115	The Specified Output directory already exists, does not exist, or cannot be reached. Specify a different directory or delete the existing directory.
120	Required OUTDIR specification not found or was incorrectly placed in setup (.SEQ) file.
130	Required default files not found in setup (.SEQ) file.
990	Steps in SEQ File exceeds the maximum allowable number of steps, which is 99.
999	Unrecognized keyword in setup file.
1000 + n	A control file was missing, incorrectly placed in the setup file for iteration *n,* or has an improper filename extension.
2000 + n	An environment file was missing, incorrectly placed in the setup file for Step *n,* or has an improper filename extension.
3000 + n	The Holiday File specified in the setup file for Step *n* does not exist, or has an improper filename extension.
4000 + n	The Time-Of-Use File specified in the setup file for Step *n* does not exist, or has an improper filename extension.
5000 + n	The Season File specified in the setup file for Step *n* does not exist, or has an improper filename extension.
5500 + n	The file specified with FROMSTEP option does not exist for execution in Step *n.*
6000 + n	The User-Specified Day File specified in the setup file for Step *n* does not exist, or has an improper filename extension.
6100 + n	The Record Definition File was missing in the setup file for Step *n,* or has an improper filename extension.
6200 + n	The Generation Units File was missing in the setup file for Step *n,* or has an improper filename extension.

Return Code	Description
6300 + n	The Customer Attribute File was missing in the setup file for Step *n,* or has an improper filename extension.
6400 + n	The Peak Day File was missing in the setup file for Step *n,* or has an improper filename extension.
6500 + n	The Input Keys File was missing in the setup file for Step *n,* or has an improper filename extension.
7000 + n	A Database File was missing, incorrectly placed in the setup file for step *n,* or has an improper filename extension.
7100 + n	Invalid Database Type was specified in the setup file for Step *n,* or has an improper filename extension.
7200 + n	A Rates File was missing or incorrectly placed in the setup file for Step *n,* or has an improper filename extension.
7300 + n	A Titles File was missing, incorrectly placed in the setup file for Step *n,* or has an improper filename extension.
7400 + n	A Billing Cycle File was missing, incorrectly placed in the setup file for Step *n,* or has an improper filename extension.
7500 + n	A Demand File was missing, incorrectly placed in the setup file for Step *n,* or has an improper filename extension.
7600 + n	An autoeditor RULes File was missing or incorrectly placed in the setup file for Step *n.*
7700 + n	The Supplied Date File was missing, incorrectly placed in the setup file for Step *n,* or has an improper filename extension.
8000 + n	The APPlication File was missing or incorrectly placed in the setup file for Step *n.*
8100 + n	The program file was missing or incorrectly placed in the setup file for Step *n.*
8200 + n	The Program Parameters File was missing or incorrectly placed in the setup file for Step *n.*
8300 + n	The Program Parameters File is required for this step but no PPM value was found in the setup file for Step *n.*
8400 + n	The specified DBPARMS.INF File was missing or incorrectly placed in the setup file for Step *n.*
8500 + n	The required datefile (SUPPDATE.DAT) was not found for Step *n.*
8600 + n	The required Rate Schedule parameter was missing or incorrectly placed for RUNRS execution in Step *n.*
9000 + n + - rc	Execution of Oracle Utilities Load Analysis program failed or exceeded specified stop code in Step *n.* The "- rc" following this return code is the actual return code from the program executed in Step *n.*
9100 + n	No program to run was specified in the setup file for Step *n.*
9200 + n	The stop value in the setup file for Step *n* was not numeric.

Return Code	Description
9400 + n	The file specified in the PARMS parameter of Step *n* was not found.
9600 + n	The file specified in the PARMS parameter of Step *n* was not found and is required to run CSMODENV.
9700 + n	Required parameter for CSMODENV was not specified in the setup file for Step *n*.
9800 + n	Sequencer was unable to create a required file for Step *n*. Please contact the Oracle Utilities Helpline.

Chapter Seven

The Repeater

This chapter describes the Repeater. The Repeater makes it possible to run the Standard Load Analysis, Ratio Analysis, Domains, or 100% Sample Analysis program multiple times — but you only have to submit the job once. For each pass of the program, you can vary the analysis parameters, the set of cuts to be analyzed, or both.

Note: The Program Sequencer (see **Chapter Six**: **The Program Sequencer** also provides the ability to run programs multiple times, and is the recommended method for running multiple analyses.

This chapter includes:

- **Using the Repeater to Run Program Multiple Times**

Using the Repeater to Run Program Multiple Times

The Repeater is a convenient feature that enables you to apply the Standard Load Analysis (Y310), Ratio Analysis (Y330), Domains Analysis Programs (Y350/Y360), or 100% Sample Analysis (Y380) program multiple times in one job submission. This means that you can apply an analysis program a number of times with different parameters and/or a different sets of cuts—without having to re-submit the job for each variation.

This section includes:

- **Control and Environment Files**
- **Return Codes**

Control and Environment Files

You specify each set of cuts to be analyzed in a Control File, and you specify the analysis parameters to be applied to it in a matching Environment File. (The specifications for the contents of these files are the same as the process you are running, such as Y310, regardless of whether or not you are using the Repeater; see the *Oracle Utilities Load Analysis Load Data Analysis User's Guide* for details.)

Control files must be identified with the ".CTL" extension. For each Control File, you must supply a matching Environment File that contains the analysis parameters to be applied to the data in the corresponding Control File. The Environment File must have the same name as its corresponding Control File, with an "e" appended to the filename. The Environment File must be identified with the ".ENV" extension. So, if a Control File was named MYCUTS.CTL, the Environment File associated with it must be named MYCUTSE.ENV.

You must put all of the Control and Environment files for the Repeater run in one folder located on the server. *When you submit the job, the Repeater automatically runs the analysis program as many times as there are Control File/Environment File pairs in the folder.* This folder must be located on the server in the COMMON\REPEATER\<application-id> folder. For example, if you intend to run multiple analysis for all rate classes for Ratio Analysis (Y330), you must place all control/environment file pairs that you have created in the folder:

```
<your server install folder>\COMMON\REPEATER\Y330
```

All users should have access to this folder.

The application identifier of the analysis program (Y310, Y330, Y350, Y360, or Y380) must be used as the name of the folder containing the file pairs.

Note: Add the Server setting "RPTR_RENAME_ICS" setting with a value of "CTL" to the Server configuration run-time file (CSLSTAR.GLB) to cause the Repeater rename the ICS file in the output folders to conform to the associated control file for that iteration. This helps avoid having all the ICS files named the same.

The Repeater also accepts an unmatched environment file, one for each Analysis program and located in the same folder as the Control and Environment files for each application ID that is used to contain common environment file commands used for all runs. This is called a repeater Base Environment file (BNV). This file may contain common commands, such as PEAk or DATe, used for each iteration. When using a base environment file, you would remove the command contained in the file from the Matched Environment files. When the repeater runs, it will append to all environment files prior to program execution the contents of the Base Environment file. It will NOT check to see if there are already the same commands in the file it is modifying. The Oracle Utilities Load Analysis Graphical User Interface will show all unmatched .ENV files in the appropriate repeater folder in the "Base Environment File" selection list. Selecting "None" will run the repeater using solely the content of each matched environment file.

How to Run the Repeater:

1. From the Oracle Utilities Load Analysis desktop, go to the **Submit Panel** for the analysis program that you want to run in the Repeater mode.

   The following programs can be used in this mode:

   Standard Load Analysis (Y310), Ratio Analysis (Y330), Domains Analysis Programs (Y350/ Y360), and 100% Sample Analysis (Y380).

2. Click the **Repeater Disabled** button. It will display **Repeater Enabled** to show that the Repeater is on.

3. Select the databases that you want to apply the job to.

4. Select a Control File. (This is not required to successfully submit a job.)

   If the control file is associated with an environment file, that file will appear in the Environment File field. If it does not have an associated environment file the field will display "none." This is an indication that you do not have a matching environment file for that control file. You may choose to edit any of these files.

5. This step is optional. Select a Base Environment (BNV) File.

   The Base Environment File drop-down list appears when the repeater is enabled. The list displays all environment files in the repeater folder that are not matched to a control file. The contents of the BNV file will be appended to the environment files for each step. You may choose to edit this file.

6. Click **File Preview** if you want to preview the selected environment, control, and base environment files.

7. Click the **Submit** button.

   The **Job Parameters** panel appears.

8. Review the results (if you need additional information about using the Oracle Utilities Load Analysis Graphical User Interface to examine the output, see **Chapter Three**: **Working with Jobs** in this guide).

   If an error occurs in one of the analysis passes, Oracle Utilities Load Analysis creates a log file (REPTRERR.LOG) and places it in the root job directory. Even if it encounters errors, the job continues to run to the end.

   The Repeater groups the output of each program run in subfolders under the job folder (the name of the job folder was specified in the Job Name field in the Job Parameters dialog box illustrated above). You may need to click the Refresh Results button on the toolbar to see these folders. Each subfolder is named according to the Control File used in the run. For example, if the Control File was TGY31aa.ctl and the Job Name was Y3302041, the output for the run would be found in <job drive>\Y3302041\TGY31AA.

   > **Note**: To see the job directories, double-click on the job name for that Repeater run.

Return Codes

The following return codes signal an error. These codes can appear in the **Queue** window or the output reports.

Return Code	Description
99	The Analysis program failed for one or more Control files
100099	Could not open Control File
100100	Syntax error
100101	Error condition
100102	Halt condition (ctl-c)
200000+	Maximum return code from Oracle Utilities Load Analysis program indicates that there were either Control or Environment files that could not be found. Check the REPTRERR.LOG in the root job directory for details.

Appendix A

Programs Available via Oracle Utilities Load Analysis

This appendix lists each of the Oracle Utilities Load Analysis programs that you can use via the Oracle Utilities Load Analysis graphical user interface, including a brief description of each program's application and where you can go to find detailed operating instructions for it.

Category	Program	Application	Reference
Input	AXDB Update (X180)	Specify automatic modifications to incoming customer interval data; used in conjunction with Direct input and Production Input programs.	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 7
	Direct Input (X110, Y130)	Input .LSE format interval data from a local drive to a Oracle Utilities Load Analysis database	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 5
	Manual Entry (X120, Y220)	Input manually-created data files into a Oracle Utilities Load Analysis database	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 5
	Production Input (X111, Y131)	Input .LSE format interval data from a network file server to a Oracle Utilities Load Analysis database	Oracle Utilities Load Analysis User's Guide, Chapter Five
Validation	Cut Series Validation (X210)	Check newly-entered, newly-edited, or user-specified cuts in the CLDB to ensure that the data is accurate and error-free	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 11
	Invalid Series Validation (X220)	Generate a key list of invalid cuts	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 14 (see X810)
Editing	Delete Cuts (Q91D)	Delete a cut and all its associated records from a Oracle Utilities Load Analysis database	Oracle Utilities Load Analysis User's Guide, Chapter Five
	Load Data Editor (X310, Y630)	Modify cuts stored in either the CLDB or the ELDB	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 9
	Load Data Editor Syntax Scan (X320)	Check edit commands prior to editing the cuts.	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 9
Reporting	AXDB Summary (X170)	Report contents of the AXDB	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 7
	Cut Series Gap Reporter (X490, X491, Y490, Y491)	Identify gaps in interval data	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 16 (see X490)
	Cut Series Overlap Reporter (X530, Y530, X531, Y531)	Identify overlaps in interval data	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 16 (see X530)
	Late Cut Reporter (X470)	Identifies missing data by comparing cut stop-dates to a specified point in time	Oracle Utilities Load Analysis Load Data Analysis User's Guide (Optional Extension)

Category	Program	Application	Reference
	Load Data Reporter (X410, X420, Y420, Y430)	Report contents of selected customer load data records or computed statistics (X410 for CLDB, X420 for ALDB, Y420 for ELDB, Y430 for SLDB)	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 12
	Summary Reporter (X440, X460, Y440, Y460)	Report summary information about records in the database (X440 for CLDB, X460 for ALDB, Y440 for ELDB, Y460 for SLDB)	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 12
	Time Series Reporter (X400, Y410)	Report selected cuts or statistics in tabular format (columns and rows)	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 12
	Totalizing Reporter (X430, Y450)	Report load profile data and summary statistics for multiple channels of data	Oracle Utilities Load Analysis Load Data Management User's Guide (Optional Extension)
	Validation Statistics Reporter (X480)	Applies four internal validation tests to input data and reports summary statistics on failing cuts	Oracle Utilities Load Analysis Load Data Management User's Guide (Optional Extension)

Category	Program	Application	Reference
Retrieval	Copy Cuts (Q91C)	Copy data from one Oracle Utilities Load Analysis database to another	Oracle Utilities Load Analysis User's Guide, Chapter Five
	Load Data Retrieval (X660)	Copy cuts from the ALDB back to the CLDB, allowing you to re-examine and revise them as needed	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 15
	Data Retrieval (Y960)	Copy cuts from the SLDB back to the ELDB, allowing you to re-examine and revise them as needed	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 17
Data Extraction	Billed Energy (Y230)	Compute energy use from demand; create control files for Analysis programs	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 13
	Direct Output (X720, X740, Y720, Y740)	Output Oracle Utilities Load Analysis data for use with Interval Data Manager and non-Oracle Utilities programs	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 16
	Load Data Extraction (Y240)	Copy load data from the ALDB and/or CLDB into the ELDB to be used with the Oracle Utilities Load Analysis Analysis programs	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 5

Category	Program	Application	Reference
	Statistical Package (X710, Y710)	Output load data in file formats compatible with third-party software file formats	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 16
Analysis	100% Sample Analysis (Y380)	Compute load statistics for 100% sampled populations, such as large industrials	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 8
	Aggregate Load Analysis (Y320)	Combine individual rate class statistics into estimates for major customer classes and total system load	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 9
	Coincident Peak Analysis (Y340)	Estimate the mean and corresponding sampling error for up to 12 periods of coincident peak demand for a customer class.	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 10
	Day Type Analysis (X760, X770, Y760, Y770)	Produce 24-hour average hourly load profiles for up to 48 daytypes in a year.	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 18
	Domains Analysis MPU (Y350)	Compute statistics for one or more subpopulations in an existing sample, using mean per unit expansion	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter Two
	Domains Analysis Ratio (Y360)	Compute statistics for one or more subpopulations in an existing sample, using the "combined estimate" technique	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter Two
	Individual Customer Analysis (X370, Y370)	Compute time-of-use, entire period, and average day statistics and reports	Oracle Utilities Load Analysis Load Data Analysis User's Guide
	Ratio Analysis (Y330)	Compute class- and strata- level statistics for sample customer data using ratio expansion	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 7
	Standard Load Analysis (Y310)	Compute class- and strata- level statistics for sample customer data using mean per unit expansion	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 7
Query	Key Generator (X810, X820, Y810, Y820)	Create lists of cut keys or edit blocks that match a set of user-defined criteria	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 13

Category	Program	Application	Reference
Administrator	Customer Record Definition (Y120)	Create a file that contains a table describing the format of the Customer Attribute File.	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 14
	Move Cuts (Q91M)	Move a cut and all associated records from one database to another.	Oracle Utilities Load Analysis User's Guide, Chapter Five
	Scan Archive/Delete (X910)	Move cuts from the CLDB to the ALDB, making cuts available for analysis	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 15
	Archive/Delete (Y910)	Move cut statistics from the ELDB to the SLDB, making cuts available for analysis	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 17
Transformation	Load Data Transformation (X620, Y620)	Calculate new data from existing data (e.g. apply loss factors, compute KVA, or combine multiple channels of data) in CLDB and ELDB	Oracle Utilities Load Analysis Load Data Analysis User's Guide, Chapter 15
Other (COSI)	Allocate Sampling Error (G210)	Allocates unexplained sampling error among rate classes	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Eleven
	Allocate T&D Losses (G250)	Allocate transmission and distribution losses and unbillable loads to rate classes	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Twelve
	Cost Allocators (G610)	Develop cost allocation factors	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Thirteen
	Direct Input (G110)	Input interval data into GLDB	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Six
	Direct Output (G720)	Output data from GLDB in Oracle Utilities or EEI format	Cost of Service Interface User's Guide, Chapter Fifteen
	Editor (G310)	Modify cut keys and cut data in the GLDB	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Ten
	Energy Allocators (G130)	Develop interval data from total energy values and input to GLDB	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Nine
	Rates Reporter (G410)	Report load data and statistics at each level of data development in COSI	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Fourteen
	ELDB/SLDB Extraction (Z120)	Extract data from ELDB or SLDB and inputs into GLDB	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Seven

Category	Program	Application	Reference
	Statistics Output (G710)	Output statistics from GLDB for use in other programs, such as spreadsheets	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Sixteen
	Summary Reporter (G440)	List contents of the GLDB or RLDB and database statistics	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Fourteen
	Time Series Reporter (G430)	Report user-selected statistics or cuts in tabular format	Oracle Utilities Load Analysis Cost of Service Interface User's Guide, Chapter Fourteen
	Key Generator (G810)	Create lists of cut keys or edit blocks that match a set of user-defined criteria	Oracle Utilities Load Analysis Load Data Management User's Guide, Chapter 18
Other (Sampling)	Multidimensional Population Analysis (B220)	Create sample cells and assign customers to them	Oracle Utilities Load Analysis Sampling User's Guide, Chapter Six
	Multidimensional Sample Design (B320)	Determine sample size for a multidimensional sample design	Oracle Utilities Load Analysis Sampling User's Guide, Chapter Eight
	Multidimensional Sample Selection (B420)	Draw a list of customers for participation in a multidimensional sample design	Oracle Utilities Load Analysis Sampling User's Guide, Chapter Ten
	Population Data File Conversion (B960)	Convert a population data file	Oracle Utilities Load Analysis Sampling User's Guide
	Record Definition (B110)	Define the record format of the Population Data File so other Sampling programs can read it	Oracle Utilities Load Analysis Sampling User's Guide, Chapter Three
	Sample Validation (B520)	Validate the sample selection for a multidimensional design	Oracle Utilities Load Analysis Sampling User's Guide, Chapter Eleven
	Single Dimensional Population Analysis (B210)	Analyze population frequency distribution	Oracle Utilities Load Analysis Sampling User's Guide, Chapter Four
	Single Dimensional Sample Design (B310)	Define state boundaries and determine the sample size of a single dimensional sample design	Oracle Utilities Load Analysis Sampling User's Guide, Chapter Five
	Single Dimensional Sample Selection (B410)	Draw a list of customers for participation in a single dimensional sample design	Oracle Utilities Load Analysis Sampling User's Guide, Chapter Nine
Other	Proxy Day Selection (X670)	Finds a past day that most closely matches a target date in terms of user-specified criteria	Oracle Utilities Load Analysis Load Data Analysis User's Guide
	Sequencer (SEQR)	Runs a user-specified sequence of programs	Oracle Utilities Load Analysis User's Guide, Chapter Seven

Appendix B

Oracle Utilities Load Analysis Naming Conventions

This appendix lists each of the conventions used to identify programs and files in Oracle Utilities Load Analysis, including

- **Input and Output Files**
- **Programs**
- **Templates**

Input and Output Files

The following three-letter file name extensions are used in Oracle Utilities Load Analysis to identify input and output files:

Input Files

File Type	File Extension	Required?
Environment	.env	Yes
Control	.ctl	Yes
Common > Holiday (user-editable)	.hol	Yes
Common > Time-of-Use (user-editable)	.tou	Yes
Common > Season (user-editable)	.sea	Yes
Common > Demand Period (user-editable)	.dem	Yes
Common > User-Specified Days (user-editable)	.usd	Yes
Common > Billing Cycle (user-editable)	.cyc	Yes
Common > Peak Days (user-editable)	.pea	Yes
Common (read-only) (user-editable)	.inf	Yes
Direct Input	.inp	Yes
Direct Input	.lse	Yes
Local Input	.inp	Yes
Production Input	.inp	Yes
Production Input	.lse	Yes
Sequencer Setup File	.seq	Yes

Output Files

File Type	File Extension	Required?
Reports	.rep	No. System Administrator can assign a different name in the CSLSTAR.GLB configuration file.
Data	.dat .kys .srt	No No No
Run Logs	.log	No. System Administrator can assign a different name in the CSLSTAR.GLB configuration file.

Programs

In Oracle Utilities Load Analysis, the following conventions are used to identify programs:

Program for...	Convention
Data Management Subsystem	X*xxx* where *xxx* is the program number*. For example, and **X410** identifies the **Load Data Reporter** for the CLDB.
Load Analysis Subsystem	Y*xxx* where *xxx* is the program number*. For example, **Y330** identifies the **Ratio Analysis** Program.
Oracle Utilities Load Analysis Administration	Q*xxx* where *xxx* is the program number*.0 For example, **Q91D** identifies the **Delete Cuts** Program.
Cost of Service Interface Subsystem	G*xxx* **or** Z*xxx* where *xxx* is the program number*. For example, **G210** identifies the **Allocate Sampling Error** Program, and **Z120** identifies the **ELDB/SLDB Extraction** Program.
Sampling Subsystem	B*xxx* where *xxx* is the program number*. For example, **B520** identifies the **Sample Validation** Program.

* The *Oracle Utilities Load Analysis Quick Reference Guide* offers a comprehensive list of all program names and numbers.

Templates

Oracle Utilities Load Analysis recognizes Environment File and Control File templates by the following file names:

Template	Convention
For Control Files	TG*xnn*A.tem where *x* is the Oracle Utilities Load Analysis subsystem (D or X for Load Data Management and E or Y for Load Analysis) and *nn* is the first two digits of the program number*. For example, TGY33A.tem is the template for Ratio Analysis Control files in the Load Analysis Subsystem.
For Environment Files	TG*xnn*B.tem where *x* is the Oracle Utilities Load Analysis subsystem (D or X for Load Data Management and E or Y for Load Analysis) and *nn* is the first two digits of the program number*. For example, TGY33B.tem is the template for Ratio Analysis Environment files.

* The *Oracle Utilities Load Analysis Quick Reference Guide* offers a comprehensive list of all program names and numbers.

Template files are located in the Common\Data folder on the server machine.

Appendix C

Oracle Utilities-Supplied Input Files

This appendix lists each of the test input files that Oracle Corporation supplies with Oracle Utilities Load Analysis. You can use these files as a starting point for your own work.

When delivered by Oracle Utilities, they are named according to the conventions used to identify templates (TG*xnn*A.tem and TG*xnn*B.tem). That means they will appear when you click the **Template** button in the **File Composer** window. (See **Templates** on page B-4 for an explanation of the naming conventions for templates.)

Category	Program Name	Control File	Environment File	Other Input
Input	AXDB Update (X180)	TGX18A	TGX18B	
	Direct Input for .LSE format (X110, Y130)	NA	TGY13B, TGX41C	TGX11E (load data) TGY13E
	Manual Entry (X120, Y220)	TGY22A	NA	
	Production Input for .LSE format (X111, Y131)	NA	TGY13B	TGX11E (load data) TGY13E (load data)
Validation	Cut Series Validation (X210)	TGX21A	TGX21B	
	Invalid Series Validation (X220)	TGX22A		
Editing	Delete Cuts (Q91D)	TGQ91DA	TGQ91DB	
	Load Data Editor (X310, Y630)	TGX31A	TGX31B	
	Load Data Editor Syntax Scan (X320)	TGX32A	TGX32B	
Reporting	AXDB Summary (X170)	NA	TGX17B	
	Late Cut Reporter (X470)	TGX47A	TGX47B	
	Load Data Reporter (X410, X420, Y420, Y430)	TGX41A, TGX42A, TGY42A, TGY43A	TGX41B, TGY42B, TGX42B, TGY43B	TGY31C, TGY31D, TGY31E
	Summary Reporter (X440, X460, Y440, Y460)	TGX44A	TGX44B	
	Time Series Reporter (X400, Y410)	TGY41A	TGY41B	
	Totalizing Reporter (X430, Y450)	TGX43A	TGX43B	
	Validation Statistics Reporter (X480)	TGX48A		
	Cut Series Gap Reporter (X490, X491, Y490, Y491)		TGX49B	

	Cut Series Overlap Reporter (X530, X531, Y530, Y531)	TGX53A	TGX53B	
Retrieval	Copy Cuts (Q91C)	TGQ91CA	TGQ91CB	
	Data Retrieval (Y960)	TGY96A	TGY96B	
	Load Data Retrieval (X660)	TGX66A	TGX66B	
Data Extraction	Billed Energy (Y230)	TGY23A	TGY23B	
	Direct Output for .LSE format (X720, X740, Y720, Y740)	TGY72A	TGY72B	
Category	**Program Name**	**Control File**	**Environment File**	**Other Input**
	Statistical Package (X710, Y710)	TGX71A, TGY71A	TGX71B, TGY71B	
	Load Data Extraction (Y240)	TGY24A	TGY24B	
Analysis	Aggregate Load Analysis (Y320)	TGY32A	TGY32B	TGY31C, TGY31D, TGY31E
	Coincident Peak Analysis (Y340)	TGY34A	TGY34B	
	Day Type Analysis (X760, X770, Y760, Y770)	TGX76A, TGX77A, TGY76A, TGY77A	TGX76B, TGY76B, TGX77B, TGY77B	TGY31C, TGY31D
	Domains Analysis MPU (Y350)	TGY35A	TGY35B	TGY31C, TGY31D, TGY31E, TGY31F
	Domains Analysis Ratio (Y360)	TGY36A	TGY36B	TGY31C, TGY31D, TGY31D, TGY31F
	Individual Customer Analysis (X370, Y370)	TGX37A, TGY37A	TGY37B, TGY37B	TGY31C, TGY31D, TGY31E, TGY23E, TGY37E
	Ratio Analysis (Y330)	TGY33A	TGY33B	TGY31C, TGY31D, TGY31E, TGY31F
	Standard Load Analysis (Y310)	TGY31A	TGY31B	TGY31C, TGY31D, TGY31E, TGY31F
	100% Sample Analysis (Y380)	TGY38A	TGY38B	TGY31C, TGY31D, TGY31E

Query	Key Generator (X810, X820, Y810, Y820)	TGX81A	NA	
Administrator	Move Cuts (Q91M)	TGQ91MA	TGQ91MB	
	Archive/Delete (Y910)	TGY91A	NA	
	Customer Record Definition (Y120)	TGY12A	NA	
	Scan Archive/Delete (X910)	TGX91A	TGX91B	
Transformation	Load Data Transformation (X620, Y620)	TGX62A, TGY62A	TGX62B, TGY62B	TGY31C, TGY31D
Other (COSI)	Allocate Sampling Error (G210)	NA	TGG21B	TGG21C
	Allocate T&D Losses (G250)		TGG25B	TGG21C, TGE31D, TGE31C
	Cost Allocators (G610)		TGG61B	TGG61C, TGG21C, TGE31D, TGE31C, TGG61D
	Direct Input (G110)		TGG11B	
	Initialize GLDB/RLDB (Q943)	NA		
	Direct Output (G720)	TGG72A	TGG72B	
	Editor (G310)	TGG31A		

Category	Program Name	Control File	Environment File	Other Input
	ELDB/SLDB Extraction (Z120)	TGZ12A	TGZ12B	
	Energy Allocators (G130)	TGG13A	TGG13B	
	GLDB Key Generator (G810)	TGG81A		
	Rates Reporter (G410)		TGG41B	TGG41C, TGG21C, TGE31D, TGE31C
	Statistics Output (G710)		TGG71B	TGG21C, TGE31D
	Summary Reporter-GLDB (G440)	TGG44A	TGG44B	
	Summary Reporter-RLDB (G450)		TGG45B	
	Time Series Reporter (G430)	TGG43A	TGG43B	
Other (Sampling)	Multidimensional Population Analysis (B220)	TGB22A		SCDB, TGB22C
	Multidimensional Sample Design (B320)		TGB32B	TGB32A
	Multidimensional Sample Selection (B420)	TGB42A		SCDB, TGB22C, TGB22A
	Record Definition (B110)	TGB12A		SCDB
	Sample Validation (B520)			TGB52A, TGB52B, TGB52C
	Single Dimensional Population Analysis (B210)	TGB21A		SCDB, TGB22C
	Single Dimensional Sample Design (B310)		TGB31B	TGB31A
	Single Dimensional Sample Selection (B410)	TGB41A		SCDB, TGB22C, TGB22A
Proxy	Proxy Day Selection (X670)		TGX67B	

Appendix D

Oracle Utilities Load Analysis Sequencer Program Input Tables and Keywords

This appendix lists the basic components of Step Definition Records for each Oracle Utilities Load Analysis program that you can include in a SEQ File. (See **Chapter Six**: **The Program Sequencer** for detailed information about SEQs.), including:

- **Base Keywords**
- **Load Data Management Subsystem**
- **Load Data Analysis Subsystem**
- **Cost of Service Interface**
- **Language Codes for LOCALE Setting**

Base Keywords

Keywords are case-insensitive.

Keyword	Description
CTL	ConTroL filename (required) / INPut (.INP) filenames will be accepted here as well as files ending with the extension set by the ANALMASK Command in CSLSTAR.GLB.
ENV	ENVironment filename (required)
BNV	Base eNVironment filename
HOL	HOLiday filename (optional)
TOU	Time Of Use filename (optional)
SEA	SEAson filename (optional)
USD	User Specified Day filename
KYS	Input Keys File (Key Generator)
PEA	PEAk Day File
GUF	Generating Units File (COSI)
DBL	Record Definition File
CAF	Customer Attribute File
CYC	billing CYCle filename
DEM	DEMand filename
DBA	DataBAse to use - can have several DBA entries within one block, parameter must be supplied to specify DB type (CLDB, ELDB, etc.) Format: DBA <*RDB/tablesetname*> databasetype
CRF	COSI Rates File filename (must be supplied in SEQ File when needed)
CTF	COSI Titles File filename (must be supplied in SEQ File when needed)
RUL	RULes File for X210 (must be supplied if want to trigger auto editor commands)
PGM	ProGraM to run in this step (non-Oracle Utilities) (Fully Qualified and includes any parameters to this program)
PPM	Program ParaMeters (optional) (Fully Qualified) - contains a single record of parameters to be sent to program specified by PGMs.
APP	Oracle Utilities Load Analysis Application to run. (the specific Applid of the job) ex: X110 or X420, etc.
OUT1	Output filename
STP *n*	If the result, or return code from the program exceeds or equals *n* then the stream will stop, Default = 99.

Load Data Management Subsystem

This section lists Sequence programs available for the Load Data Management subsystem.

An asterisk (*) next to a keyword means the keyword is optional, and may default.

An asterisk (*) next to a database type means the database is optional.

X110 - Direct Input - Validation - AutoEditor

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X110	
DBA	<*fully qualified databasename*>	CLDB
DBA	<*fully qualified databasename*>	AXDB
CTL	Filename (Data File)	
ENV	Filename	
RUL *	Filename	No Default

X120 - Manual Entry

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X120	
DBA	<*fully qualified databasename*>	ELDB
CTL	Filename (Data File)	

X170 - AXDB Summary Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X170	
DBA	<*fully qualified databasename*>	AXDB
ENV	Filename	

X180 - AXDB Update

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X180	
DBA	<*fully qualified databasename*>	AXDB
CTL	Filename	
ENV	Filename	

X210 - Cut Series Validation

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X210	
DBA	<*fully qualified databasename*>	CLDB
DBA	<*fully qualified databasename*>	AXDB
CTL	Filename	
ENV *	Filename	TGX21B.ENV
RUL *	Filename	No Default

X220 - Invalid Cut Series Validation

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X220	
DBA	<*fully qualified databasename*>	CLDB
DBA	<*fully qualified databasename*>	AXDB
CTL	Filename	
DBL	Filename	

X310 - Load Data Editor

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X310	
DBA	<*fully qualified databasename*>	CLDB
DBA	<*fully qualified databasename*>	AXDB
CTL	Filename	
ENV *	Filename	TGX21B.ENV
RUL *	Filename	No Default

X320 - Load Data Editor Syntax Scan

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X320	
DBA	<*fully qualified databasename*>	CLDB
CTL	Filename	
ENV*	Filename	

X370 - Individual Customer Analysis (CLDB)

Keywords:	Values or DBA Types	Default (rodfiles – datfiles)
APP	X370	
DBA	<*fully qualified databasename*>	CLDB
CTL	Filename	
ENV	Filename	
DEM*	Filename	
CYC*	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	

X400 - Time Series Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X400	
DBA	<*fully qualified databasename*>	CLDB
CTL	Filename	
ENV	Filename	

X410 - Load Data Reporter (CLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X410	
DBA	<*fully qualified databasename*>	CLDB
DBA*	<*fully qualified databasename*>	ALDB
DBA*	<*fully qualified databasename*>	ALDB2
CTL	Filename	
ENV *	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	

X420 - Load Data Reporter (ALDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X420	
DBA	<*fully qualified databasename*>	CLDB
DBA*	<*fully qualified databasename*>	ALDB
DBA*	<*fully qualified databasename*>	ALDB2
CTL	Filename	
ENV *	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	

X430 - Totalizing Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X430	
DBA	<*fully qualified databasename*>	CLDB
CTL	Filename	
ENV	Filename	
HOL*	Filename	
TOU*	Filename	

X440 - Summary Reporter (CLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X440	
DBA	<*fully qualified databasename*>	CLDB
CTL*	Filename	
ENV	Filename	

X460 - Summary Reporter (ALDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X460	
DBA	<*fully qualified databasename*>	CLDB
CTL*	Filename	
ENV	Filename	

X470 - Late Cut Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X470	
DBA	<*fully qualified databasename*>	CLDB
CTL	Filename	
ENV	Filename	

X480 - Validation Statistics Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X480	
DBA	<*fully qualified databasename*>	CLDB
CTL	Filename	

X490 - CLDB Cut Series Gap Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X490	
DBA	<*fully qualified databasename*>	CLDB
CTL*	Filename	
ENV	Filename	

X491 - ALDB Cut Series Gap Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X491	
DBA	<*fully qualified databasename*>	CLDB
CTL*	Filename	
ENV	Filename	

X530 - CLDB Cut Series Overlap Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X530	
DBA	<*fully qualified databasename*>	CLDB
CTL*	Filename	
ENV	Filename	

X531 - ALDB Cut Series Overlap Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X531	
DBA	<*fully qualified databasename*>	CLDB
CTL*	Filename	
ENV	Filename	

X620 - CLDB Load Data Transformation

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X620	
DBA	<*fully qualified databasename*>	CLDB
DBA*	<*fully qualified databasename*>	CLDB2
CTL*	Filename	
ENV	Filename	

X660 - ALDB Load Data Retrieval

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X660	
DBA	<*fully qualified databasename*>	CLDB
DBA	<*fully qualified databasename*>	ALDB
CTL*	Filename	
ENV*	Filename	

X670 - Proxy Day Selection

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X670	
DBA	<*fully qualified databasename*>	CLDB
DBA*	<*fully qualified databasename*>	CLDB2
ENV	Filename	
HOL*	Filename	
SEA*	Filename	

X710 - Statistical Package Interface

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X710	
DBA	<*fully qualified databasename*>	CLDB
CTL	Filename	
ENV	Filename	

X720 - Direct Output (CLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X720	
DBA	<*fully qualified databasename*>	CLDB, ALDB*, ALDB2*
CTL	Filename	
ENV	Filename	

X740 - Direct Output (ALDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X740	
DBA	<*fully qualified databasename*>	ALDB
DBA*	<*fully qualified databasename*>	ALDB2*
CTL	Filename	
ENV	Filename	

X760 - Day Type Analysis (CLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X760	
DBA	<*fully qualified databasename*>	CLDB
DBA	<*fully qualified databasename*>	ALDB
DBA	<*fully qualified databasename*>	ALDB2
CTL	Filename	
ENV	Filename	
PEA*	Filename	
HOL*	Filename	
SEA*	Filename	

X770 - Day Type Analysis (ALDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X770	
DBA	<*fully qualified databasename*>	ALDB
DBA*	<*fully qualified databasename*>	ALDB2
CTL	Filename	
ENV	Filename	
PEA*	Filename	
HOL*	Filename	
SEA*	Filename	

X810 - Key Generator (CLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X810	
DBA	<*fully qualified databasename*>	CLDB
CTL	Filename	
DBL	Filename	
ENV*	Filename	
KYS*	Filename	

X820 - Key Generator (ALDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X820	
DBA	<*fully qualified databasename*>	CLDB
CTL	Filename	
DBL	Filename	
ENV*	Filename	
KYS*	Filename	

X910 - Scan Archive/Delete

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X910	
DBA	<*fully qualified databasename*>	CLDB
DBA	<*fully qualified databasename*>	ALDB
CTL*	Filename	
ENV	Filename	

Load Data Analysis Subsystem

This section lists Sequence programs available for the Load Data Analysis subsystem.

An asterisk (*) next to a keyword means the keyword is optional, and may default.

An asterisk (*) next to a database type means the database is optional.

Q91M - Move Cuts

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Q91M	
DBA	<*fully qualified databasename*>	CLDB, ELDB, SLDB, ALDB
CTL	Filename	
ENV	Filename	

Q91D - Delete Cuts

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Q91D	
DBA	<*fully qualified databasename*>	ELDB. SLDB, CLDB, ALDB
CTL	Filename	
ENV	Filename	

Q91C- Copy Cuts

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Q91C	
DBA	<*fully qualified databasename*>	CLDB
CTL	Filename	
ENV	Filename	

Y120 - Customer Record Definition

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y120	
CTL	Filename	

Y130 - Direct Input (ELDB)/Enhanced Direct Input (ELDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y130	
DBA	*<fully qualified databasename>*	ELDB
DBA	*<fully qualified databasename>*	AXDB
CTL	Filename (Data File)	
ENV	Filename	
RUL*	Filename	No Default

Y220 - Manual Entry

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y220	
DBA	*<fully qualified databasename>*	ELDB
CTL	Filename (Data File)	

Y230 - Billed Energy Program

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y230	
DBA	*<fully qualified databasename>*	ELDB
CTL	Filename	
ENV	Filename	
CYC	Filename	

Y240 - Load Data Extraction

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y240	
DBA	<*fully qualified databasename*>	CLDB
DBA*	<*fully qualified databasename*>	ALDB
DBA*	<*fully qualified databasename*>	ALDB2
DBA	<*fully qualified databasename*>	ELDB
CTL	Filename	
ENV	Filename	

Y310 - Standard Load Analysis

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y310	
DBA	<*fully qualified databasename*>	ELDB
DBA	<*fully qualified databasename*>	CLDB2
BNV	Filename	
CTL	Filename	
ENV	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	
USD*	Filename	

Y320 - Aggregate Load Analysis

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y320	
DBA	<*fully qualified databasename*>	ELDB
CTL	Filename	
ENV	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	

Y330 - Ratio Analysis

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y330	
DBA	<*fully qualified databasename*>	ELDB
DBA*	<*fully qualified databasename*>	CLDB2
BNV	Filename	
CTL	Filename	
ENV	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	
USD*	Filename	

Y340 - Coincident Peak Analysis

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y340	
DBA	<*fully qualified databasename*>	ELDB
DBA	<*fully qualified databasename*>	ALDB
CTL	Filename	
ENV	Filename	

Y350 - Domain Analysis (MPU)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y350	
DBA	<*fully qualified databasename*>	ELDB
DBA*	<*fully qualified databasename*>	CLDB2
BNV	Filename	
CTL	Filename	
ENV	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
USD*	Filename	

Y360 - Domain Analysis (Ratio)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y360	
DBA	<*fully qualified databasename*>	ELDB
DBA	<*fully qualified databasename*>	CLDB2
CTL	Filename	
ENV	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	
USD*	Filename	

Y370 - Individual Customer Analysis (ELDB)

Keywords:	Values or DBA Types	Default (rodfiles – datfiles)
APP	Y370	
DBA	<*fully qualified databasename*>	ELDB
CTL	Filename	
ENV	Filename	
DEM*	Filename	
CYC*	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	

Y380 - 100% Sample Analysis

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y380	
DBA	<*fully qualified databasename*>	ELDB
DBA	<*fully qualified databasename*>	CLDB2
CTL	Filename	
ENV	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	
USD*	Filename	

Y410 - Time Series Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y410	
DBA	<*fully qualified databasename*>	ELDB
CTL	Filename	
ENV	Filename	

Y420 - Load Data Reporter (ELDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y420	
DBA	<*fully qualified databasename*>	ELDB
DBA	<*fully qualified databasename*>	SLDB
DBA	<*fully qualified databasename*>	SLDB2
CTL	Filename	
ENV*	Filename	

Y430 - Load Data Reporter (SLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y430	
DBA	<*fully qualified databasename*>	SLDB
DBA*	<*fully qualified databasename*>	SLDB2
CTL	Filename	
ENV*	Filename	

Y440 - Summary Reporter (ELDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y440	
DBA	<*fully qualified databasename*>	ELDB
CTL*	Filename	
ENV	Filename	

Y450 - Totalizing Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y450	
DBA	<*fully qualified databasename*>	ELDB
CTL	Filename	
ENV	Filename	
HOL*	Filename	
TOU*	Filename	

Y460 - Summary Reporter (SLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y460	
DBA	<*fully qualified databasename*>	SLDB
CTL*	Filename	
ENV	Filename	

Y490 - Cut Series Gap Reporter (ELDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y490	
DBA	<*fully qualified databasename*>	ELDB
CTL*	Filename	
ENV	Filename	

Y491 - Cut Series Gap Reporter (SLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y491	
DBA	<*fully qualified databasename*>	ELDB
CTL*	Filename	
ENV	Filename	

Y530 - Cut Series Overlap Reporter (ELDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y530	
DBA	<*fully qualified databasename*>	ELDB
CTL*	Filename	
ENV	Filename	

Y531 - Cut Series Overlap Reporter (SLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y531	
DBA	<*fully qualified databasename*>	ELDB
CTL*	Filename	
ENV	Filename	

Y620 - Load Data Transformation

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y620	
DBA	<*fully qualified databasename*>	CLDB
DBA*	<*fully qualified databasename*>	CLDB2
CTL	Filename	
ENV	Filename	
HOL*	Filename	
TOU*	Filename	

Y630 - Load Data Editor (ELDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y630	
DBA	<*fully qualified databasename*>	ELDB
CTL	Filename	
ENV	Filename	

Y710 - Statistical Package Interface

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y710	
DBA	<*fully qualified databasename*>	ELDB
CTL	Filename	
ENV	Filename	

Y720 - Direct Output (ELDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y720	
DBA	<*fully qualified databasename*>	ELDB
DBA	<*fully qualified databasename*>	SLDB
DBA	<*fully qualified databasename*>	SLDB2
CTL	Filename	
ENV	Filename	

Y740 - Direct Output (SLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y740	
DBA	<*fully qualified databasename*>	SLDB
DBA*	<*fully qualified databasename*>	SLDB2
CTL	Filename	
ENV	Filename	

Y760 - Day Type Analysis (ELDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y760	
DBA	<*fully qualified databasename*>	ELDB
DBA	<*fully qualified databasename*>	SLDB
DBA	<*fully qualified databasename*>	SLDB2
CTL	Filename	
ENV	Filename	
PEA*	Filename	

Y770 - Day Type Analysis (SLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y770	
DBA	<*fully qualified databasename*>	SLDB
DBA	<*fully qualified databasename*>	SLDB2
CTL	Filename	
ENV	Filename	
PEA*	Filename	

Y810 - Key Generator (ELDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y810	
DBA	<*fully qualified databasename*>	ELDB
CTL	Filename	

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
DBL	Filename	
ENV*	Filename	
KYS*	Filename	

Y820 - Key Generator (SLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y820	
DBA	*<fully qualified databasename>*	ELDB
CTL	Filename	
DBL	Filename	
ENV*	Filename	
KYS*	Filename	

Y910 - Archive/Delete

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y910	
DBA	*<fully qualified databasename>*	ELDB
DBA	*<fully qualified databasename>*	SLDB
CTL	Filename	
DBL	Filename	

Y960 - Data Retrieval

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Y960	
DBA	*<fully qualified databasename>*	ELDB
DBA	*<fully qualified databasename>*	SLDB
CTL*	Filename	
ENV	Filename	

Cost of Service Interface

This section lists Sequence programs available for the Cost of Service Interface subsystem.

An asterisk (*) next to a keyword means the keyword is optional, and may default.

An asterisk (*) next to a database type means the database is optional.

G110 - Direct Input (GLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	G110	
DBA	<*fully qualified databasename*>	CLDB
DBA	<*fully qualified databasename*>	AXDB
CTL	Filename INP or LSE	
ENV	Filename	

Z120 - Data Extraction (SLDB/ELDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	Z120	
DBA	<*fully qualified databasename*>	GLDB
DBA*†	<*fully qualified databasename*>	ELDB
DBA*†	<*fully qualified databasename*>	SLDB
DBA*	<*fully qualified databasename*>	AXDB
CTL	Filename	
ENV	Filename	

†At least one database (an SLDB or an ELDB) must be specified.

G130 - Energy Allocators

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	G130	
DBA	<*fully qualified databasename*>	GLDB
CTL	Filename	
ENV	Filename	
HOL*	Filename	

G210 - Allocate Sampling Error

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	G210	
DBA	<*fully qualified databasename*>	GLDB
DBA	<*fully qualified databasename*>	GLDB2
ENV	Filename	
CRF	Filename	
HOL*	Filename	
TOU*	Filename	

G250 - Allocate T&D Losses

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	G250	
DBA	<*fully qualified databasename*>	GLDB
ENV	Filename	
CRF	Filename	
HOL*	Filename	
TOU*	Filename	

G410 - Rates Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	G410	
DBA	<*fully qualified databasename*>	GLDB
ENV	Filename	
CRF	Filename	
CTF	Filename	
HOL*	Filename	
TOU*	Filename	

G430 - Time Series Reporter

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
DBA	<*fully qualified databasename*>	ELDB

G440 - Summary Reporter (GLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
DBA	<*fully qualified databasename*>	CLDB

G610 - Cost Allocators

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	G610	
DBA	<*fully qualified databasename*>	GLDB
DBA	<*fully qualified databasename*>	RLDB
ENV	Filename	
CRF	Filename	
CTF	Filename	
HOL*	Filename	
TOU*	Filename	
SEA*	Filename	

G670 - Proxy Day

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
APP	X670	
DBA	<*fully qualified databasename*>	CLDB
DBA	<*fully qualified databasename*>	CLDB2
ENV	Filename	
HOL	Filename	
SEA	Filename	

G720 - Direct Output (GLDB)

Keywords:	Values or DBA Types	Default (rodfiles - datfiles)
DBA	<*fully qualified databasename*>	CLDB

G810 - Key Generator (GLDB)

Keywords:	Values or DBA Types	D efault (rodfiles - datfiles)
DBA	<*fully qualified databasename*>	CLDB

Language Codes for LOCALE Setting

The LOCALE setting uses a 3-digit language code to specify the date and number formats used by the programs. Use the following table to determine the proper language code for your country/ locale:

Language (Locale)	Language Code
Afrikaans (South Africa)	AFK
Albanian (Albania)	SQI
Arabic (Algeria)	ARG
Arabic (Bahrain)	ARH
Arabic (Egypt)	ARE
Arabic (Iraq)	ARI
Arabic (Jordan)	ARJ
Arabic (Kuwait)	ARK
Arabic (Lebanon)	ARB
Arabic (Libya)	ARL
Arabic (Morocco)	ARM
Arabic (Oman)	ARO
Arabic (Qatar)	ARQ
Arabic (Saudi Arabia)	ARA
Arabic (Syria)	ARS
Arabic (Tunisia)	ART
Arabic (U.A.E.)	ARU
Arabic (Yemen)	ARY
Armenian (Armenia)	HYE
Azeri (Cyrillic, Azerbaijan)	AZE
Azeri (Latin, Azerbaijan)	AZE
Basque (Basque)	EUQ
Belarusian (Belarus)	BEL
Bengali (India)	BNG
Bosnian (Bosnia and Herzegovina)	BSB
Bosnian (Cyrillic) (Bosnia and Herzegovina)	BSC
Bulgarian (Bulgaria)	BGR
Catalan (Catalan)	CAT
Chinese (Hong Kong S.A.R.)	ZHH

Language (Locale)	Language Code
Chinese (Macao S.A.R.)	ZHM
Chinese (People's Republic of China)	CHS
Chinese (Singapore)	ZHI
Chinese (Taiwan)	CHT
Croatian (Bosnia and Herzegovina)	HRB
Croatian (Croatia)	HRV
Czech (Czech Republic)	CSY
Danish (Denmark)	DAN
Divehi (Maldives)	DIV
Dutch (Belgium)	NLB
Dutch (Netherlands)	NLD
English (Australia)	ENA
English (Belize)	ENL
English (Canada)	ENC
English (Caribbean)	ENB
English (Ireland)	ENI
English (Jamaica)	ENJ
English (New Zealand)	ENZ
English (Republic of the Philippines)	ENP
English (South Africa)	ENS
English (Trinidad and Tobago)	ENT
English (United Kingdom)	ENG
English (United States)	ENU
English (Zimbabwe)	ENW
Estonian (Estonia)	ETI
Faroese (Faroe Islands)	FOS
Filipino (Philippines)	FPO
Finnish (Finland)	FIN
French (Belgium)	FRB
French (Canada)	FRC
French (France)	FRA
French (Luxembourg)	FRL
French (Principality of Monaco)	FRM

Language (Locale)	Language Code
French (Switzerland)	FRS
Frisian (Netherlands)	FYN
Galician (Galician)	GLC
Georgian (Georgia)	KAT
German (Austria)	DEA
German (Germany)	DEU
German (Liechtenstein)	DEC
German (Luxembourg)	DEL
German (Switzerland)	DES
Greek (Greece)	ELL
Gujarati (India)	GUJ
Hebrew (Israel)	HEB
Hindi (India)	HIN
Hungarian (Hungary)	HUN
Icelandic (Iceland)	ISL
Indonesian (Indonesia)	IND
Inuktitut (Latin) (Canada)	IUK
Irish (Ireland)	IRE
Italian (Italy)	ITA
Italian (Switzerland)	ITS
Japanese (Japan)	JPN
Kannada (India)	KDI
Kazakh (Kazakhstan)	KKZ
Kiswahili (Kenya)	SWK
Konkani (India)	KNK
Korean (Korea)	KOR
Kyrgyz (Kyrgyzstan)	KYR
Latvian (Latvia)	LVI
Lithuanian (Lithuania)	LTH
Luxembourgish (Luxembourg)	LBX
Macedonian (Former Yugoslav Republic of Macedonia)	MKI
Malay (Brunei Darussalam)	MSB

Language (Locale)	Language Code
Malay (Malaysia)	MSL
Malayalam (India)	MYM
Maltese (Malta)	MLT
Maori (New Zealand)	MRI
Mapudungun (Chile)	MPD
Marathi (India)	MAR
Mohawk (Canada)	MWK
Mongolian (Cyrillic, Mongolia)	MON
Nepali (Nepal)	NEP
Northern Sotho (South Africa)	NSO
Norwegian, Bokmål (Norway)	NOR
Norwegian, Nynorsk (Norway)	NON
Pashto (Afghanistan)	PAS
Persian (Iran)	FAR
Polish (Poland)	PLK
Portuguese (Brazil)	PTB
Portuguese (Portugal)	PTG
Punjabi (India)	PAN
Quechua (Bolivia)	QUB
Quechua (Ecuador)	QUE
Quechua (Peru)	QUP
Romanian (Romania)	ROM
Romansh (Switzerland)	RMC
Russian (Russia)	RUS
Sami (Inari) (Finland)	SMN
Sami (Lule) (Norway)	SMJ
Sami (Lule) (Sweden)	SMK
Sami (Northern) (Finland)	SMG
Sami (Northern) (Norway)	SME
Sami (Northern) (Sweden)	SMF
Sami (Skolt) (Finland)	SMS
Sami (Southern) (Norway)	SMA
Sami (Southern) (Sweden)	SMB

Language (Locale)	Language Code
Sanskrit (India)	SAN
Serbian (Cyrillic) (Bosnia and Herzegovina)	SRN
Serbian (Cyrillic, Serbia)	SRB
Serbian (Latin) (Bosnia and Herzegovina)	SRS
Serbian (Latin, Serbia)	SRL
Slovak (Slovakia)	SKY
Slovenian (Slovenia)	SLV
Spanish (Argentina)	ESS
Spanish (Bolivia)	ESB
Spanish (Chile)	ESL
Spanish (Colombia)	ESO
Spanish (Costa Rica)	ESC
Spanish (Dominican Republic)	ESD
Spanish (Ecuador)	ESF
Spanish (El Salvador)	ESE
Spanish (Guatemala)	ESG
Spanish (Honduras)	ESH
Spanish (Mexico)	ESM
Spanish (Nicaragua)	ESI
Spanish (Panama)	ESA
Spanish (Paraguay)	ESZ
Spanish (Peru)	ESR
Spanish (Puerto Rico)	ESU
Spanish (Spain)	ESN
Spanish (Uruguay)	ESY
Spanish (Venezuela)	ESV
Swedish (Finland)	SVF
Swedish (Sweden)	SVE
Syriac (Syria)	SYR
Tamil (India)	TAM
Tatar (Russia)	TTT
Telugu (India)	TEL
Thai (Thailand)	THA

Language (Locale)	Language Code
Tswana (South Africa)	TSN
Turkish (Turkey)	TRK
Ukrainian (Ukraine)	UKR
Urdu (Islamic Republic of Pakistan)	URD
Uzbek (Cyrillic, Uzbekistan)	UZB
Uzbek (Latin, Uzbekistan)	UZB
Vietnamese (Vietnam)	VIT
Welsh (United Kingdom)	CYM
Xhosa (South Africa)	XHO
Zulu (South Africa)	ZUL

Appendix E

Using Oracle Utilities Load Analysis Sample Data

Oracle Utilities Load Analysis provides new users the ability to execute much of the functionality within the product using supplied data and pre-configured templates. This allows users to get accustomed to how the application works using this sample data. This appendix describes how to start using the system with this data, including:

- **Sample Data Provided**
- **Prerequisites**
- **Using The Sample Data In a Test Scenario**
    - **Loading the Sample LSE and INP Files Into a Database**
    - **Running a procedure using the sample data and the templates**

Sample Data Provided

Oracle Utilities Load Analysis includes sample data allowing users to immediately begin using the system with this data in a testing scenario. This data includes:

- Load Data (interval data in INP and LSE format)
- Template files pre-configured to match the load data
- Default schedules such as TOU, season, and holiday that matches the templates and load data

Prerequisites

Before using this sample data with Oracle Utilities Load Analysis, ensure that the following prerequisites have been met:

- Oracle Utilities Load Analysis client and server have been installed and are functioning according to the *Oracle Utilities Load Analysis Installation Guide*.
- Database connections and configurations completed and tested according to the *Oracle Utilities Load Analysis Installation Guide*.
- The client must be able to log onto the server and successfully submit a job.
- All database types (CLDB, ELDB, ALDB, AXDB, SLDB, GLDB) must be defined to the server and available for selection on the GUI.
- Standard installed template files must be unaltered located in the common\data folder on the server. (.tem files)
- Configuration files and sample data has been set up as described in **Setting up Configuration and Default files** on page 1-14 in the *Oracle Utilities Load Analysis Installation Guide*.

Using The Sample Data In a Test Scenario

The following section describes the procedure to load the test interval data into non-production, test databases and use that data to run reports, procedures, and analyses.

Loading the Sample LSE and INP Files Into a Database

The first jobs you should run are the X111 (Production Input) procedures. These procedures load the interval data contained within the formatted LSE and INP files into the CLDB, ALDB and ELDB databases. The CLDB database is a special database type that contains data that has not yet been edited and/or validated or copied to into the analysis database (ELDB). Validated data that is as complete and accurate as possible is stored in the ELDB databases. The ALDB database is an archive type database used to archive older data from the CLDB.

Use the following procedure to load sample data.

1. Navigate to the Production Input submit screen by expanding the Input branch and clicking on the X111 (production Input) selection. This will change the submit screen to allow you to select INP and LSE files, select input configurations and databases.
2. Select your test CLDB database (TEST CLDB) in the **Interval Database** drop-down list.
3. Select the SAMPLECLDB.LSE file in the **Load Data** drop-down list.
4. Check the **Combine Reports** checkbox.
5. Leave all options as is.
6. Click on the new **Environment File** button (This is the last button on the Environment file line in the center frame) to open the composer. The composer will assist you in creating a load configuration. During this exercise, we will use the template.
7. Click on the **Template** button to load the template containing "LOA REP"
8. Save the configuration to whatever filename you desire. (TSTX11B.ENV is what we will use in this example).
9. Click the Submit button to display the Job Parameters screen that names the job and submits the procedure to the server for execution.
10. Leave all fields as defaulted and click **OK**.

    This starts the execution of the procedure, creates a job folder shown in the Results frame, and adds an entry to the queue frame. If the server is configured properly the input job will run successfully. You should notice a zero return code in the queue frame once the procedure has completed.
11. Click the entry in the results frame to highlight the report in the results file folder display.
12. Open the report and view it to ensure that the data was loaded properly.
13. Repeat the procedure for the other three sample files (outlined in the table below) until all data (.lse and .inp files) has been successfully loaded into the appropriate databases.

    You will need to adjust the environment file for each file based on the table below. You may get some informational messages about DST, these messages are OK with this data unless the data is not imported.

Database Selection	Load Data File	Env options to use
CLDB	SampleCLDB.LSE	LOA REP (use template)
ELDB	SampleELDB.LSE	LOA REP VAL
GLDB	SampleGLDB.LSE	LOA REP VAL

Database Selection	Load Data File	Env options to use
GLDB	SampleGLDB.INP	LOA REP VAL DST A

Running a procedure using the sample data and the templates

You are now ready to run procedures, reports, and analysis using the test data and the supplied templates. The steps outlined below may be applied to most other Oracle Utilities Load Analysis procedures.

How to execute Load Analysis procedures:

1. Start by selecting the procedure you are interested in running from the left-hand tree selector. In this example we will be running a Ratio Analysis on the validated data located in the ELDB database using the templates.
2. Expand the **Analysis** branch on the main screen to show the analysis procedures and select **Ratio Analysis**. In the configuration and selection frame select the ELDB you just loaded the SampleELDB.LSE data.
3. Select none for the **Target Database**.
4. Create a new Control File by clicking on the **Create New File** button on the Control file line. This will open the composer for Ratio Analysis.
5. Click on the **Template** button to populate the file with the configuration that matches the sample data.
6. Close the file and save it as any filename you wish.
7. Repeat steps 3 - 5 to create a new Environment file. There are two examples in this template. You must select one and comment it out prior to saving the file. You do this by selecting the example and clicking on the comment button. (The second configuration is commented out.)
8. Leave all other selections as "default".
9. Click on the **Submit** button to run the procedure.
10. View the analysis results and graphs.

Index

T

U

W