



License Tracker

... maximizing value by understanding usage

LT-Analyzer Administration Manual

Version 3.3

May 31 2011

Contents

LT Analyzer 3.3 Administration Manual	11
Document History	11
Legal Notifications	13
Trademarks	13
Copyright	13
About this Document	15
Purpose of this Document	15
Conventions used in this Document	16
Fundamental Concepts	17
Log Files	17
Content Scopes	18
Report Scopes	18
Filters	19
Models	19
User Model	21
Feature Model	22
License Sharing Strategy	23
False Denial Threshold	23
Application Type	24
Product Name	24
Feature Alias	24
Session Source Model	24
License Models	25
Budget Models	28
Types of Users	28
Accounts	29
Token Based Reporting	29
Project Reporting	30
Global Licenses	30
Editions of LT Analyzer	30
Server DataBase Edition	30
File DataBase Edition	31
Limited Edition	31
Installing <i>LT Analyzer</i>	33

How to Install <i>LT Analyzer</i>	33
Getting a Valid <i>LT Analyzer</i> License	34
"License not for this release" Problem	34
Setting Up a Server Database	35
Setting Up a MySQL Server Database	35
How To Set Up a MySQL Server Database	35
Install the MySQL Server	36
Customize the MySQL Server Configuration File for <i>LT Analyzer</i> Operation	36
How to Create a <i>LT Analyzer</i> Database	38
How to Create an <i>LT Analyzer</i> Administrator in a MySQL Database	38
How to Create <i>LT Analyzer</i> Users in a MySQL Database	39
How to Create <i>LT Analyzer</i> Tables in a MySQL Server Database	40
Necessary Information for Users	40
How to Upgrade the MySQL Database from a Previous <i>LT Analyzer</i> Database version	41
Setting Up an Oracle Server Database	42
How to set up an Oracle Server Database	42
Creating Roles for Read-write and Read-only Users	43
Creating the Schema Owner	43
Creating the <i>LT Analyzer</i> Database Tables	44
Defining the Privileges for the Roles	44
Creating a <i>LT Analyzer</i> Administrator in an Oracle Database	45
Creating <i>LT Analyzer</i> Users in an Oracle Database	45
Necessary Information for Users	45
Upgrading the Oracle Database from a Previous <i>LT Analyzer</i> Database version	46
Setting Up a New File Database	47
Prerequisites	47
How to set up a File Database	47
Managing Usage Data	51
Prerequisites	51
Opening an Existing Server Database	51
Opening an Existing File Database	51
Importing License Usage Data	52
Exporting Information from a Database	54
Editing Session Information	55
Deleting Sessions	55
How to Delete Sessions	55
Deleting Import Sources	56
How to Delete Session Sources	56

Setting System Preferences	56
How to set system preferences	56
Managing Models	59
The Model Manager	59
Model Management Dialog	59
Standard Editing Procedures	60
How to prepare a model for editing	60
How to Edit a Row	61
How to Edit More than one Row at a Time	62
Working with the User Model	63
Importing a User Model file	63
How to Edit the User Model	63
How to Add a New User	64
How to Edit an Existing User	64
How to Delete a User	65
How to Add an Item	65
How to Edit an Existing Item	66
How to Delete an Item	66
How to Add a User Category	67
How to Add or Edit Attributes Under a User Category	67
How to Delete a User Category	67
Working With LDAP	68
Connecting to an LDAP Server	68
Importing User Data from LDAP	69
Editing the Feature Model	69
How to Add a New Feature	70
How to Edit an Existing Feature	70
How to Delete a Feature	71
How to Assign a Feature Alias	71
Working with the License Model	72
Importing a License Model	72
Editing a License Model	72
How to Add a New License	72
How to Edit an Existing License	73
How to Delete License Items	74
Working with the Account Model	74
Creating an Account Model	74
Editing an Account Model	75
Adding Users to an Account Model	75

Editing a Budget Model	75
How to Add a Year to a Budget Model	76
How to Edit an Existing Budget Model	76
Managing Scopes	77
The Scope Manager	77
Scope Management Dialog	78
Editing a Content Scope	78
Adding a New Content Scope	78
How to Delete a Content Scope	80
Editing a Report Scope	81
How to Add a New Report Scope	81
How to Edit a Report Scope	81
How to Delete a Report Scope	82
Automating <i>LT Analyzer</i>: AutoTracker	83
AutoTracker Database Specification	83
Using AutoLoader	83
Command Line Syntax	84
Preliminaries and Context for the Script	85
Required Items	85
Options	86
Notes	87
Examples	88
Using AutoReporter	88
Command Line Syntax	88
Preliminaries and Context for the Script	89
Required Items	89
Options	90
Notes	91
Using the AutoAlerter	92
Command Line Syntax	92
Preliminaries and Context for the Script	92
Required Items	93
Options	93
Input Directory	94
AlertModel CSV file	94
Long Session Alerts Syntax	94
Denial Alert	95
Subscription Model	96

Support and Contact Information	96
Support Services	96
Contact Information	97

Tables

Table 1: LT Analyzer Supported Log File Formats	17
Table 2: Effects of Changes to my.ini	37
Table 3: LT Analyzer Log File Formats and Extensions	52
Table 4: AutoLoader Log File Formats and Extensions.....	84
Table 5: Long Session Alert Entry in the AM.csv file.....	94
Table 6: Denial Alert Entry in the AM.csv file	95
Table 7: A Subscription Model File	96

LT Analyzer 3.3 Administration Manual

This is the administrator's manual for LT Analyzer 3.3.

End-user information is available in the *LT Analyzer 3.3 User Manual*.

Document History

LT Analyzer 3.3 May 2011

- Minor Release
- Introduction of Account Model
- Integration with LDAP
- Database Upgrade
- Interface Upgrades

License Tracker 3.2, October 2009

- Minor Release
- Introduction of Token-Based Reporting
- Introduction of Enhanced Debug Log (.enh)
- Introduction of attribute to feature model
- Introduction of Project Reporting
- Introduction of Global Licenses
- Introduction of Session Source Model

License Tracker 3.1 August 2008

- Minor release.
- Introduction of time-variable end-user attributes.
- Introduction of tree view and sort of data.

License Tracker 3.0 November 2007

- Major release.
- Introduction of MySQL and Oracle server databases and administration.
- Administration capability removed from end-users.

License Tracker 2.8, January 2007

- Minor release.

Improvements to reports and charts. Support for new log formats.

Legal Notifications

This section details various legal notifications.

Trademarks

IBM LUM™, Flexlm™, License Tracker™, LT Analyzer™, MySQL™, Oracle™, Petrel™, Novell™, Reprise™, Windows™, Solaris™, Linux™, MS-Access™, and Sentinel™ are product names that are mentioned for identification purposes only and are trademarks of their respective owners.

Copyright

Copyright © 2005, 2006, 2007, 2008, 2009, 2010, 2011 License Tracker Inc. All rights reserved.

No part of this document may be reproduced, stored in an information retrieval system, translated or retransmitted in any form or by any means, electronic or mechanical, including photocopying and recording, without prior written permission from the copyright owner, License Tracker Inc. See <http://www.licensetracker.ca> for current contact information.

About this Document

This section describes this manual.

Purpose of this Document

This document is written for administrators of *LT Analyzer* and *LT Analyzer* MySQL and Oracle license usage databases.

Topics include:

- installing *LT Analyzer*,
- setting up a database on a server or local file database,
- controlling access to *LT Analyzer* administrator functions,
- importing session data into the database,
- deleting selected session data,
- deleting complete session logs,
- managing
 - the user model,
 - the feature model,
 - the license models,
 - the budget models
- managing content scopes,
- managing report scopes,
- setting various thresholds for reporting denials, and
- setting up and running Auto Tracker.

End-user information, including detailed instructions on

- selecting content scopes,
- creating charts and reports,
- using the filter capability,

- using Web Tracker, and
- general user interface interaction,
- managing the session source model

can be found in the *LT Analyzer 3.3 Users' Manual*.

Conventions used in this Document

This document follows these conventions:

- Buttons and menu items are referenced in bold font, for example, **File**, or displayed as a screen capture.
- Menu paths are also displayed in bold font, for example, **File > Create File DB...**
- References to the *LT Analyzer* application, the *Web Tracker* module, the *Auto Tracker* module, the *Auto Tracker Alerts* module and the *LT Analyzer Manual* are displayed in italics.
- In the sections that follow, references to the *LT Analyzer* application, the *Web Tracker* module, the *Auto Tracker* module and the *Auto Tracker Alerts* module are shortened to *LT Analyzer*, *Web Tracker*, *Auto Tracker* and *Auto Tracker Alerts* respectively.
- Clickable links such as cross references and Internet addresses are in blue font.
- Critically important information that affects the proper functioning of the software or the report generated is prefaced by the word **IMPORTANT**. For example:
***IMPORTANT:** An example of IMPORTANT information.*
- Clarifications and reminders are generally prefaced by the word **NOTE**. For example:
***NOTE:** An example of a note or clarification.*
- Graphic changes between minor releases are marked as **UPDATED**. New user interface elements are marked as **NEW**. Graphic changes between the last minor release and a new major release are not flagged in any way.
- User input, generally on the command line or in a dialog is in Courier font. For example
`This is text entered in a dialog or the command line.`

Fundamental Concepts

This section provides an overview of the fundamental concepts that must be understood to use *LT Analyzer*.

LT Analyzer analyzes a variety of software usage log formats generated by most high-value software packages. These log files are typically generated by license managers used to control access to the software packages, or in some cases they are directly generated by the software packages themselves.

LT Analyzer processes the supported log files and inserts the usage data into a *LT Analyzer* database.

Once the log information is captured in a database in a standard format, *LT Analyzer* can analyze the data further and generate useful cost and license usage information. This information, captured in reports and charts, can then be used for controlling and reviewing license usage and cost.

Users and administrators need to understand several key concepts to make effective use of the capabilities built into *LT Analyzer*. These concepts include log files, content scopes, report scopes, filters, models, administrator versus end-users and server databases versus file databases.

Log Files

Software license usage is recorded in log files (session login and logout times, denial of access, server startup and other events) on a continual basis. These data must be imported into *LT Analyzer* before any usage analysis can be performed. The supported log formats are:

Table 1: LT Analyzer Supported Log File Formats

Format
FLEXlm™ Debug Log
FLEXlm™ Enhanced Debug Log
FLEX RIF Report
Reprise™ Report

Format
Petrel™ Usage Report
Petrel™ SLOG Report
IBM LUM™ Report
Sentinel™ Log
License Tracker™ ASCII Log
License Tracker™ Binary Log

Content Scopes

The concept of scopes is fundamental to getting the most value from *LT Analyzer* and *Web Tracker*.

From a technical point of view a content scope is simply a view into the database. It allows administrators to specify exactly the data that should be extracted and analyzed to produce reports and charts. In *LT Analyzer*, a content scope is a fairly coarse definition of a subset of the whole database.

Administrators define scopes for end-users. Typically a content scope will be used to provide the end-user access to all of license information for a particular location (Houston or Calgary), a management unit (Division 1, Division 2), or a particular vendor (Vendor A, Vendor B) or some combination of categories and attributes for users and features.

Defining a set of content scopes users will want to use has a significant impact on the load that will be placed on a server database. If every end-user has access to a useful content scope, then every report generated will require searching a dataset much smaller than the whole database.

Once an end-user selects a predefined content scope that best applies to them, the data can be further refined by the end-user with filters.

Report Scopes

Report scopes are used to specify the contents for a set of Web Tracker reports. They are used primarily as a control mechanism when using the Auto Reporter function of Auto Tracker to generate many (potentially different) sets of Web Tracker reports.

From a technical point of view a **report scope** is simply a definition of the contents for a set of Web Tracker output. It allows administrators to specify which of types of reports (feature, user, group and analyses) should be produced, as well as which components of the selected reports are to be produced. The **report scope** also allows an administrator to specify which user categories are to be included in the group reports and the group analysis component of the feature reports.

Defining a set of report scopes recognizes that different information is needed for various usage analysis requirements, and more importantly that certain reports are not needed for many purposes. Typically an organization will create a **report scope** specifying all reports and then create other **report scopes** reducing the number of reports generated (and therefore the computation time required to generate the set of Web Tracker reports) for specific purposes.

Once an end-user selects a predefined report scope that best applies to their Web Tracker reporting needs, they can further refine the report set definition in the Web Tracker report selection dialog.

Filters

Filters, like content scopes, also define a subset of data but at a finer level of detail. Filters are generally applied to data that has already been limited by a content scope defined by the administrator.

End-users generally select a scope that best applies to their situation, for example, a location such as “Calgary”. Within this content scope they can filter the data to define finer grained report criteria, for example specific users, groups, vendors, or features.

This combination of content scope and filter accomplish two objectives simultaneously:

- 1) Filters allow end-users complete freedom in defining exactly what information appears in their reports, based on any criteria available in the data.
- 2) Content scopes allow administrators to define commonly used views into the database.

End-users can apply filters to the entire database if they wish. If end-users do this frequently it may indicate that there is a mismatch between the existing content scopes and end-users reporting requirements. The easiest way to determine what the problem is to talk to end-users who apply filters to the entire database.

Models

LT Analyzer makes extensive use of

- user,

- feature,
- license
- budget, and
- session source

models in its interaction with software usage databases.

These models either:

- Add attributes to data captured from log files that can be used to refine a query (for example, Business Unit and Role associated with userIDs) or to control a computation (for example, a false denial threshold).
- Add information to data captured from log files that enhances understanding of the raw data in reports (for example, what is the first name, last name and telephone number for each userID).

All of these models are managed by the *LT Analyzer* administrator. They are important because they determine to a large extent what can appear in a *LT Analyzer* report.

Creating an *LT Analyzer* database usually involves collecting and assembling information in a user model, feature model, license models and budget models as well as importing the log files that will provide the basic data for the *LT Analyzer* reports and charts.

Strictly speaking, only the log files are required before generating reports, but most organizations will want to access the reports based on information captured in the user model, feature model, license models and budget models.

The models can be prepared in other tools and imported as comma-separated value (CSV) files or created and modified with the *LT Analyzer* editors. The user model, feature model, license model and budget model files can be edited from within *LT Analyzer* -- this is the normal way of editing these files. However, .csv files can be built using other tools and then imported into *LT Analyzer*. This capability may be useful when setting up reporting for large numbers of users and groups.

You can obtain the information for a model from a computer-account-management system by exporting a .csv file containing the required information, and then edit the file in a spreadsheet program to ensure the correct column order and headings.

You can also create a model file from an existing database using the export option, for example, **File > Export > User Model...** Models created in one database can be exported for use in creating other databases in the future.

User Model

The user model is a simple ASCII `.csv` file. It links a userID, called a **User Name**, found in the license usage data to a user's

- first name,
- last name,
- e-mail address, and
- phone number,
- and two built-in comment fields.

All of these fields in the user model provide information which is displayed in the header of the *Web Tracker user reports*.

The default user model contains the following headings on the first line of the ASCII file. These heading names are reserved and significant to *LT Analyzer*. Heading order is not important in the model.

User Name

The computer login name (that is, the userID) found in the license server log files. `UserName` is the only required heading in a user model.

First Name

The real first name of the user.

Last Name

The real last name of the user.

Account Name

The name of the account that this user belongs to.

Email Address

The user's e-mail address.

Phone Number

The user's phone number.

Comment 1

A comment about the user.

Comment 2

A comment about the user.

Date

The date from which the user attributes specified in this row are valid.

Other headings may be added to the model by an administrator. These are treated as category names by *LT Analyzer*. For example:

- **Category_1 ... Category_n**

Categories (also known as group types) correspond to extra, administrator-defined columns in the **user model**. Category names usually correspond to something significant to the organization tracking license use.

Each cell in a category column is an attribute (also known as a group) that can be used to further refine categorization. For example, the category “Location” might be subdivided with the attributes “Calgary”, “Houston”, and “Denver”.

A sample user model is found at `samples/sample_UM.csv` in the *LT Analyzer* software folder. The sample contains all of the headings from the default model and some category columns with attributes entered in the cells. Example categories included in the sample **user model** file are “Location”, “Division”, and “Role”.

User models are most often generated by exporting information from an LDAP or other account management system to a CSV file. Some modification of the headings may be required to make them match *LT Analyzer* user model requirements. Starting with version 3.3 of *LT-Analyzer*, it is possible to create connections to an LDAP system in order to automatically import user information into the user model.

There is one user model per *LT Analyzer* database.

Feature Model

The feature model is a simple ASCII `.csv` file. It specifies features (or software packages) found in the log files. Each row in the Feature Model corresponds to one feature, and the column headings are as follows:

Vendor Name

The name of the software vendor that licenses the software feature.

Feature Name

The name of the feature that the row represents

Sharing Strategy

The license sharing strategy used with this feature.

False Denial Threshold

The false denial threshold for this feature.

Application Type

The type of application this feature falls into.

Product Name

The name of the product that this feature is part of.

Feature Alias

The feature alias is used in place of the feature name in the interface, reports and charts.

Additional details on the column headings are provided below.

License Sharing Strategy

The sharing strategy specifies under what conditions *LT Analyzer* should deem multiple user sessions from the log file are using the same license. The license sharing strategies are:

- **No sharing** - There is no license sharing. Each session uses up a license.
- **Host** - If multiple concurrent or overlapping sessions are using a feature from the same host, then only one license is used.
- **User** - If a software feature is being used by concurrent or overlapping session the same user running on the same or different hosts, then only one license is used.
- **User Plus Host** - If a feature is being used by multiple concurrent or overlapping sessions of the same user on the same host, then only one license is used.

Most often, computation intensive features have no sharing while interactive software frequently has user+host sharing (same user at the same computer may have multiple sessions of the software using the same license). Any sharing information provided in the license manager log files will override those specified in the feature model.

False Denial Threshold

Removing denials caused by license server chains or high frequency failed requests can be particularly useful. The false denial threshold is used to control this. A duplicate denial (same feature, same user, same host) found within the threshold of another denial is deemed to be false; also, a denial for which a license is granted (same feature, same user, same host) within the threshold is also deemed false.

Application Type

A software feature can be organized by its function or role. This is the application type. For example, a software feature like Microsoft WordTM is a word processor, so the administrator could set the ApplicationType in the feature model to "WordProcessor". The application type can then be used when creating reports to focus on a particular type of application.

Product Name

Software products are sometimes divided into different features which are licensed separately. For example, a product that uses plugins might have each plugin licensed separately. In this case, the overall product is the Product in the feature model, and the individual plugins are the features. The Product Name is the name of the overall product that a feature is part of.

Feature Alias

Depending on the practices of the software vendor, the software features may have technical or confusing names in the log files. The Feature Alias is used to remove this confusion for *LT Analyzer* users and administrators. The feature alias is assigned in the feature model and is associated with a feature. The alias should be set to a unique, clear name and will replace the feature name in the user interface, reports and charts. This removes confusion for users and administrators.

NOTE: Note that in the feature model csv, the Feature Alias column is simply called 'Alias'.

Other headings may be added to the model by an administrator. These are treated as category names by *LT Analyzer*. For example: Category_1 ... Category_n. Categories correspond to extra, administrator-defined columns in the feature model. Category names usually correspond to something significant to the organization tracking license use. Each cell in a category column is an attribute that can be used to further refine categorization.

A sample feature model is found at `samples/sample_FM.csv` in the *LT Analyzer* software folder. The sample contains all of the headings from the default model and some category columns with attributes entered in the cells.

There is one feature model per *LT Analyzer* database.

Session Source Model

The Session Source Model provides details of each log file imported into a *LT Analyzer* database. It has default headings:

Name

The name of the file loaded.

Type

The format of the log file.

Vendor

The name of the vendor whose usage data was in the file.

Import Date

The date the file was loaded into the *LT Analyzer* database.

Start Date

First date in the log file.

End Date

Last date in the log file.

Sessions Read

Number of sessions read from the file

Sessions Loaded

Number of sessions actually loaded (may have been filtered out due to date, length, etc.).

Server

Name of the license server that this log file came from.

Other headings may be added to the model by an administrator. These are treated as category names by *LT Analyzer*. For example:

- Category_1 ... Category_n

Categories correspond to extra, administrator-defined columns in the session source model.

Each cell in a category column is an attribute that can be used to further refine categorization. For example, the category “ServerLocation” might be subdivided with the attributes “Kuala Lumpur”, “Calgary” and “London”.

License Models

The license model is a simple ASCII `.csv` file. It links feature names found in the log files to

- vendor names,
- the number of licenses owned,

- the ownership cost of licenses (average purchase and maintenance cost per license), and
- software rental details
 - rental rates: yearly, monthly, weekly, daily, or hourly;
 - rental mode: (by the second, hourly, capped, etc.) and
 - minimum duration of sessions that are to be considered when calculating rentals.

Multiple license models can be created for different financial analyses of the software usage data.

NOTE: License models are associated with content scopes and will not be considered in any analyses until such an association is created. A single license model may be associated with multiple content scopes.

A license model contains the headings listed below on the first line of the ASCII file. Only the **Feature Name** and **License Count** headings are required. If more than one line are in a license model for a single feature then they must have different valid “from” dates.

Vendor Name

The vendor's name. This is only used for adding new features found in the license model to the feature table of the *LT Analyzer* database. If the feature is already in this table then the contents of this field are ignored.

Feature Name

The software feature name as found in the license server log files.

Date

This is the date from which the data on this line is valid. This is to support the fact that license counts and costs change over time. Use the format YYYY-MM-DD.

License Count

The number of licenses purchased (owned, not rented).

Product Cost

The average cost to purchase a single license.

Maintenance Cost

The average annual maintenance fee for a single purchased license.

Annual Cost

The annual rental license fee.

Monthly Cost

Monthly rental license fee.

Weekly Cost

Weekly rental license fee.

Daily Cost

Daily rental license fee.

Hourly Cost

The hourly rental license fee. If rental billing is by the second, provide the hourly rate (PerSecondRate x 3600) in this column.

Mode

The type of rental mode. The rental mode, options are:

- NONE
- SESSION
- ANNUAL
- MONTHLY
- DAILY
- HOURLY
- HOURLY_CAPPED. If total hourly rentals exceed monthly rental costs then a monthly rental is added and hourly rentals are recalculated; superseded by HR_CAP_MON.
- HR_CAP_MON. If total hourly rentals for a single rented license exceed monthly rental costs then it is deemed to be rented for the month and hourly rentals are recalculated.
- SECOND
- SC_CAP_MON. If billing by the second for a single license exceeds the monthly rate then the license is deemed to be rented by the month and by the second rentals are recalculated.

Minimum Duration

This is minimum session length (in seconds) to be considered when calculating rental costs. All sessions shorter in duration will be ignored.

NOTE: The last six columns in the license model file refer to rental fees (annual, monthly, weekly, daily and hourly rates), and minimum duration. The mode indicates which rental model should be used by LT Analyzer. If you do not use rental licenses, then leave these columns blank in your file.

A sample license model is found at `samples/sample_LM.csv` in the *LT Analyzer* software folder.

Budget Models

The budget model is a simple ASCII `.csv` file. It is a simple list of the budgeted rental costs per month in an organization; if your organization does not have software rental arrangements with any of your software vendors then you will not need to create or use budget models.

Multiple budget models can be created for specific views of the software usage data. A typical example is a budget model for each budgeting unit in a corporation, or for rental budgets for specific software vendors.

NOTE: Budget models are associated with content scopes and will not be considered in any analyses until such an association is created. A single budget model may be associated with multiple content scopes.

A budget model must contain the following headings on the first line of the ASCII `.csv` file:

Year

The year of the data.

Month

The actual calendar month name.

Amount

The money allotted to license usage for each month.

A sample budget model is found at `samples/sample_BM.csv` in the *LT Analyzer* software folder.

Types of Users

All users have administrative privileges for file databases.

Users who do have administration privileges on the *LT Analyzer* database do have access to the **Admin** menu.

From *LT Analyzer* **Admin** menu administrators can:

- import and delete software usage data,
- create and manage user, license, budget and feature models, and
- create and manage report scopes and content scopes for the organization.

Administrators can also use *LT Analyzer* as an end-user themselves.

LT Analyzer administrator privileges are also required to configure and run Auto Tracker.

Accounts

Accounts are used to organize which users are charged, and how. Users are charged for use of software based on which account they belong to. Each account has its own license model which describes how the licenses and charges are to be applied to users in that account. A user does not have to belong to an account, but only users that belong to an account will be included in usage charges - any users not in an account will not be charged. Users can belong to only one account, or none.

Charges are calculated for each user only when session usage data is imported into the *LT Analyzer* database. If the accounts are edited or changed (by the administrator, for example) the account charges will be out of date. In this case, the transaction data will need to be deleted from the *LT Analyzer* database and reloaded.

Token Based Reporting

Traditionally, software vendors have used a feature-based licensing system where the customer purchases or rents a certain number of per-user licenses for each software feature provided by the vendor.

This model creates the situation where at certain stages of business activity, one feature might run out of licenses while another does not, creating a situation where the business must pay for more licenses of one feature while those of an already paid for feature are not used.

A token-based licensing system differs from a feature-based system in that instead of buying licenses for each user, the business buys a certain number of tokens that go into a 'token pool'. The software vendor then defines how many tokens each software feature is worth, and when that software feature is in use by any user, the corresponding number of tokens is removed from the pool. When the user stops using the software, the tokens are added back into the pool. When a user tries to use a piece of software, the token pool is checked to see if there are enough tokens. If so, the user is allowed to use the software, if not, the user is denied.

LT Analyzer version 3.2 and above allows for the analysis and tracking of this sort of license model. Typically with a token model, it is only possible to analyze the usage of the tokens. With *LT Analyzer* it is possible to track the actual usage of each software feature within a token pool.

Project Reporting

Frequently software usage is associated with a specific project within an enterprise. Whenever this information is provided in a log file *LT Analyzer* will capture this information and store it in its database. This information can then be used to divide software costs among projects within the enterprise.

Global Licenses

Software vendors can purchase a **Global License** of *LT Analyzer*. This license allows the vendors to use the *LT Analyzer* toolset within their organization and to provide *LT Analyzer* to their customers.

In order to make use of a Global License, a customer must download a **Vendor-Specific License**, which is provided at no cost from the License Tracker website. The Vendor Specific License is associated with a software vendor that has purchased a Global License, and allows a customer to install and use *LT Analyzer* to analyze license usage data from that vendor, and that vendor only. Multiple vendor specific licenses can be downloaded and installed, allowing the customer to analyze data from multiple vendors using a single *LT Analyzer* database.

Enterprises making use of Vendor-Specific Licenses have complimentary access to the User Manual and the Administrator Manual; access to all other support capabilities (such as User Forum and regularly schedule webinars) requires a separate support contract.

Editions of LT Analyzer

This section describes the editions of *LT Analyzer* available for different data usage requirements.

Server DataBase Edition

The Server DB Edition of *LT Analyzer* maintains its databases by interacting with a database server. The two databases that are supported are: Oracle 10 and MySQL 5. The database server and the *LT Analyzer* software may be running on the same machine, or they may be running on separate machines. A separate license is required for the database server. The Server DB Edition may be used on all platforms supported by the Java Runtime Environment 1.5 (i.e., Windows, Solaris, and Linux).

The Server DB Edition is appropriate for large organizations that need to analyze usage data measured in the hundreds of thousands to millions of user sessions.

File DataBase Edition

The File DB Edition of *LT Analyzer* maintains its databases in MS-Access files. The files are always created and used within the filesystem that is accessible from the user's computer. A separate copy of MS-Access is not required. The File DB Edition can only be used with Windows systems.

The File DB Edition is appropriate for organizations with small or moderate usage analysis volumes measured in the 10,000s to low 100,000s of user sessions.

Limited Edition

The Limited Edition of *LT Analyzer* is a special version of the File DB Edition. This is the default mode of *LT Analyzer* when started without a *LT Analyzer* license. The Limited Edition restricts all databases to a maximum of 10 users and 15 software features. In addition, the Limited Edition does not provide access to the Auto Tracker capabilities.

The Limited Edition is appropriate for small organizations with very low software usage. It may also be used for evaluation of the *LT Analyzer* product.

Installing *LT Analyzer*

LT Analyzer is easy to install, just ensure that your system is sufficient for the application and follow the installation procedure.

How to Install *LT Analyzer*

PREREQUISITE The minimum system requirements for running *LT Analyzer* are:

- Windows XP or Windows 7 operating system when accessing file databases (32 bit version of *LT Analyzer* only),
- Windows XP, Windows 7, Linux or Solaris operating systems when accessing server databases,
- 1 GB of RAM, 2 GB is suggested,
- Java run-time environment, version 1.5 or higher, and
- a screen resolution of 1024 X 768 pixels or greater.

To install *LT Analyzer* follow this procedure:

TASK

1. Access the *LT Analyzer* web site www.licensetracker.ca and download the zip file onto your hard drive.
2. Extract the following folders from the zip file into a *LT Analyzer* directory:
 - `bin` contains executable files (extension `.jar`), startup scripts and shortcut files.
 - `common` contains images used in the *LT Analyzer* interface. Users should not modify any files in this directory.
 - `conf` this is where you install the license file (extension `.ltilic`).
 - `doc` contains the files for the *LT Analyzer* Admin Manual and User Manual (with extension `.pdf`).
 - `samples` contains sample model and demo files.
3. Install the `<your_license_filename>.ltilic` file in the `conf` folder. This file was e-mailed to you as part of the download and evaluation or purchase process.

4. To start *LT Analyzer*, open the `bin` directory and double-click on:
 - the `LicenseTracker.lnk` shortcut for Windows users
 - the `LicenseTrackerWOW64.lnk` for users running 32-bit Java or who want to open or create a Microsoft Access database.
 - the `LicenseTracker.sh` script file for Linux/Solaris users.
-

Getting a Valid *LT Analyzer* License

Part of the *LT Analyzer* download process is requesting a demonstration or regular license. You should have received the appropriate license by e-mail from `licensetracker.ca` before installation.

If you did not receive an e-mail with the appropriate license, please contact `sales@licensetracker.ca`.

As of version 2.7, the *LT Analyzer* license (`<your_license_filename>.ltilic` in the `conf` directory) is tied to the version number of the software. A new license is required for each new version of the *LT Analyzer* software.

"License not for this release" Problem

If *LT Analyzer* reports an "invalid license" problem when the software is started after an update, follow this procedure.

TASK

1. Reinstall the license file received from *LT Analyzer* as part of the upgrade process. Make sure that you have copied the correct `<your_license_filename>.ltilic` file to the `conf` directory.
STEP RESULT: If the problem persists go to the next step.
 2. Reinstall the software as described in [“How to Install *LT Analyzer*”](#) on page 33.
STEP RESULT: If the problem persists go to the next step.
 3. Contact *LT Analyzer* support. [“Support and Contact Information”](#) on page 96.
-

Setting Up a Server Database

The normal installation of *LT Analyzer* requires a MySQL or Oracle database on a server to hold the license usage data. A separate license and support agreement is required for your selected database server and is beyond the scope of your agreement with License Tracker Inc.

This section describes how to create and populate the *LT Analyzer* database in both MySQL and Oracle.

Setting Up a MySQL Server Database

How To Set Up a MySQL Server Database

PREREQUISITE Before setting up the MySQL server, you should:

- 1) Have basic administration familiarity with MySQL including starting, stopping, and restarting a MySQL database server.
- 2) Have full administration access to the server machine.

Follow the steps in this section to create a MySQL database for *LT Analyzer*.

TASK

1. Select the computer system for the database server (ensure that this system has sufficient memory and computation power to handle queries for the scale of data you intend to store and analyze).
2. Install the MySQL 5 server software. See [“Install the MySQL Server”](#) on page 36
3. Customize the MySQL server configuration file for *LT Analyzer* operation. See [“Customize the MySQL Server Configuration File for LT Analyzer Operation”](#) on page 36.
4. Start the MySQL daemon (or Service under Windows).
5. Create the *LT Analyzer* database. See [“How to Create a LT Analyzer Database”](#) on page 38.
6. Create the *LT Analyzer* database tables. See [“How to Create LT Analyzer Tables in a MySQL Server Database”](#) on page 40

7. Create the *LT Analyzer* administrator user if necessary. See “[How to Create an LT Analyzer Administrator in a MySQL Database](#)” on page 38.
 8. Create reporting-only database user(s). See “[How to Create LT Analyzer Users in a MySQL Database](#)” on page 39
 9. Pass necessary information on to users. See “[Necessary Information for Users](#)” on page 40.
-

Install the MySQL Server

Use this procedure to install the MySQL server.

NOTE: If a MySQL server instance already exists, you may go to directly to step three (3) of this procedure.

TASK

1. Go to <http://dev.mysql.com/downloads/> and download a MySQL server (community or enterprise) installation package, version 5 or higher.
2. Start the installation wizard.
 - In this document we assume that all of the defaults have been accepted during the installation process.
 - Set the password to something you will remember. In this document we assume the password is set to `lt_admin_pw`.

STEP RESULT: The database server installation completes normally.

Customize the MySQL Server Configuration File for *LT Analyzer* Operation

Use this procedure to customize the MySQL server configuration file for use with *LT Analyzer*.

TASK

1. Choose an option file appropriate for your system and application from those provided in the main MySQL installation directory, and copy it to "my.ini" (Windows) or "my.cnf" (Unix) in the same directory.
2. Open this newly-created my.ini (Windows) or my.cnf file (Unix) file with a text editor.
3. Add these lines in the `[mysqld]` section of the `my.ini` (Windows) or `my.cnf` file (Linux/Solaris) file.

```
[mysqld]
```

```
# LTI - DEBUGGING
log_slow_queries
long_query_time = 1
# LTI PRODUCTION
innodb_file_per_table
datadir="C:/<INSTALL_LOCATION>/MySQL/<MySQL Server Folder>/Data/"
query_cache_size=64M
query_cache_limit = 8M
```

ADDITIONAL INFORMATION: These options enable correct interaction between the database server and *LT Analyzer*. The query cache size and limit values are the minimum recommended for *LT Analyzer* operation.

NOTE: Substitute the appropriate path for *<INSTALL LOCATION>* and *<MySQL Server Folder>* above.

4. Save the `my.ini` (Windows) or `my.cnf` file (Linux/Solaris) file.
5. Restart the database server in order to force it to pick up the changes in the `my.ini` (Windows) or `my.cnf` file (Linux/Solaris) file.

RESULT:

The changes to the `my.ini` file have the effect described in Effects of Changes to `my.ini`.

Table 2: Effects of Changes to `my.ini`

my.ini value	Effect
<code>log_slow_queries</code>	Creates a log file that captures any query that takes a long time. This is useful to debug performance issues.
<code>long_query_time = 1</code>	Sets the threshold for long queries to 1 second.
<code>innodb_file_per_table</code>	Creates one file for each table in the InnoDB data directory. This allows the administrator to manage disk space more effectively.
<code>datadir="C:/MySQL_INSTALL_LOCATION/MySQL/MySQL Server 5.0/Data/"</code>	Name of the folder to hold the database files.

my.ini value	Effect
query_cache_size=64M	Sets the cache size to 64M. This makes repetitive queries very efficient. This value should be set to the largest size possible given the limits of RAM on the server.
query_cache_limit = 8M	Sets the maximum size of any single query. This should be set large enough to hold the time consuming large queries, but not too large relative to the query_cache_size. If set too large this value will restrict the number of queries held in cache to too small a number.

How to Create a *LT Analyzer* Database

LT Analyzer requires its own database. Follow these steps to create one.

TASK

1. Start the MySQL command line tool.
2. Type `mysql -u root -p`
3. Enter the root user's mysql password at the password prompt.
4. To create an *LT Analyzer* database, type the following command: `create database <New License Tracker Database Name>;`

ADDITIONAL INFORMATION: Replace <New *LT Analyzer* Database Name>, above, with the desired name for the new database.

How to Create an *LT Analyzer* Administrator in a MySQL Database

If an administrator other than the database server administrator will be handling the *LT Analyzer* database, the server administrator needs to create a *LT Analyzer* database administrator or userID. Use this procedure to create an *LT Analyzer* administrator in a MySQL database.

TASK

1. Start the MySQL command line tool.
2. Type `mysql -u root -p`
3. Enter the root user's mysql password at the password prompt.

4. Type the following commands:

```
GRANT DELETE, INSERT, SELECT, UPDATE ON  
<LT_Analyzer_Database_Name>.* TO '<Admin User Name>'@'localhost'  
IDENTIFIED BY '<Admin User Password>';GRANT DELETE, INSERT,  
SELECT, UPDATE ON <LT_Analyzer_Database_Name>.* TO '<Admin User  
Name>'@'%' IDENTIFIED BY '<Admin User Password>';
```

ADDITIONAL INFORMATION: Replace <LT_Analyzer_Database_Name> with the name of the MySQL database, and replace <Admin User Name> and <Admin User Password> with the desired username and password for the user being created.

Note that these commands must be entered exactly as written.

How to Create *LT Analyzer* Users in a MySQL Database

The *LT Analyzer* database administrator needs to create a MySQL user ID for every user who will access the database on the server through *LT Analyzer*. The steps below describe creating a single user. Repeat step #3 as needed for additional users.

TASK

1. Type `mysql -uroot -p`
2. Enter the root user's mysql password at the password prompt.
3. Pick a name and a password for the user you are creating; in this example, "user" and "userpw" are used.

4. Type the following command:

```
GRANT SELECT ON  
<License_Tracker_Database_Name>.* TO '<License Tracker User  
Name>'@'localhost' IDENTIFIED BY '<License Tracker User  
Password>';
```

ADDITIONAL INFORMATION: Replace <LT_Analyzer_Database_Name> with the name of the LT Analyzer MySQL database. Replace <LT Analyzer User Name> and <LT Analyzer User Password> with the desired username and password for the user being created.

Note that this command must be entered exactly as written.

5. Type the following command: `GRANT SELECT ON <License_Tracker_Database_Name>.* TO '<License Tracker User Name>'@'%' IDENTIFIED BY '<License Tracker User Password>';`

ADDITIONAL INFORMATION: Replace `<LT_Analyzer_Database_Name>` with the name of the LT Analyzer MySQL database. Replace `<LT Analyzer User Name>` and `<LT Analyzer User Password>` with the desired username and password for the user being created.

Note that this command must be entered exactly as written.

How to Create *LT Analyzer* Tables in a MySQL Server Database

Before you can populate the *LT Analyzer* database, tables have to be created to hold the software usage data. Follow this procedure to create the required tables.

TASK

1. Type `mysql -uroot -p`
2. Enter the root user's mysql password at the password prompt.
3. Type `use <LT_Analyzer_Database_Name>;`

ADDITIONAL INFORMATION: Replace `<LT_Analyzer_Database_Name>`, above, with the name of the database.

4. To create the required *LT Analyzer* tables, copy and paste the text in the file `/<LT Analyzer Installation Folder>/conf/MYSQLscript.txt` onto the MySQL command line or run the contents of the file as a script.

ADDITIONAL INFORMATION: Replace `<LT Analyzer Installation Folder>` with the path to the folder where *LT Analyzer* is installed.

STEP RESULT: All of the tables required by *LT Analyzer* are created in the database.

Necessary Information for Users

All *LT Analyzer* users (both administrator and reporting-only) will need to be provided with:

- The name of the server where the MySQL database resides,
- The port number used to access the database,
- A database name,
- Their user ID and password.

It is the responsibility of the administrator to provide this information.

How to Upgrade the MySQL Database from a Previous *LT Analyzer* Database version

To upgrade an earlier version *LT Analyzer* database to the current version, use the following procedure.

TASK

1. Type `mysql -uroot -p`
 2. Enter the root user's mysql password at the password prompt.
 3. Type `use <LT_Analyzer_Database_Name>;`
ADDITIONAL INFORMATION: Replace `<LT_Analyzer_Database_Name>` with that database name, above.
 4. To determine the current database version, type: `SELECT databaseVersion FROM LTVersion;`
 5. To upgrade the *LT Analyzer* database, copy and paste the text in the file `<LT Analyzer Installation Folder>/conf/MYSQL-UPDATE-<old_version_number>-<old_version_number + 1>-script.txt` onto the MySQL command line or run the contents of the file as a script.
ADDITIONAL INFORMATION: Replace `<LT Analyzer Installation Folder>` with the path to the folder where *LT Analyzer* is installed.
Replace `<old_version_number>` with the version number found in step 4, and `<old_version_number + 1>` with the next version number.
STEP RESULT: All of the tables in the *LT Analyzer* database have been upgraded to the next version.
 6. Repeat the previous step as necessary with increased version numbers until you have upgraded to the latest version
-

Setting Up an Oracle Server Database

How to set up an Oracle Server Database

PREREQUISITE

- 1) Basic administration familiarity with Oracle including starting, stopping, restarting, creating tablespaces, and administering users on an Oracle database.
- 2) Basic familiarity with creating and using tables and creating and using scripts.
- 3) Basic familiarity with running SQLplus, the Oracle command line tool.
- 4) The administrator must have full administration access to the server machine and the database server.

Follow the steps in this section to create an Oracle database for *LT Analyzer*.

TASK

1. Select the Oracle server and instance in which you want to place the *LT Analyzer* database. Ensure that this system has sufficient memory and computation power to handle queries for the scale of data you intend to store and analyze.
2. Create a tablespace for the *LT Analyzer* database.
 - Ensure the initial size is adequate, suggested minimum size is 100MB.
 - Ensure that autoextend is enabled.
 - For purposes of the rest of the examples in this section we will assume that a tablespace 'ltrack' has been created.
3. Create roles for read-write and read-only. See [“Creating Roles for Read-write and Read-only Users”](#) on page 43
4. Create the schema owner. See [“Creating the Schema Owner”](#) on page 43
5. Create the *LT Analyzer* database tables. See [“Creating the LT Analyzer Database Tables”](#) on page 44
6. Define the privileges for the two roles. See [“Defining the Privileges for the Roles”](#) on page 44
7. Create administrator database user(s). See [“Creating a LT Analyzer Administrator in an Oracle Database”](#) on page 45

8. Create reporting-only database user(s). See “[Creating LT Analyzer Users in an Oracle Database](#)” on page 45
 9. Pass necessary information on to users. See “[Creating LT Analyzer Users in an Oracle Database](#)” on page 45.
-

Creating Roles for Read-write and Read-only Users

LT Analyzer identifies administrator and view-only users of an Oracle *LT Analyzer* database by which of the read-write or read-only roles they have had assigned to them by the Oracle DBA.

You may call these roles by whatever name you choose; for clarity they will be called LT_RW and LT_RO for the rest of this section.

To create the required roles, follow his procedure.

TASK

1. Log into the desired Oracle instance as the DBA (root user)
2. In the SQLplus command window type:

```
create role LT_RO not identified;  
create role LT_RW not identified;
```

Creating the Schema Owner

To create the schema owner follow this procedure.

TASK

1. Select the name for your *LT Analyzer* schema instance (suggested default is 'lt').
2. Log into the desired Oracle instance as the DBA (root user).
3. In the SQLplus window type:

```
create user lt identified by <lt_passwd>  
temporary tablespace temp  
default tablespace ltrack
```

```
quota unlimited on ltrack;  
grant create session, connect, resource to lt;
```

ADDITIONAL INFORMATION: If the schema owner will be logging into the database as a *LT Analyzer* administrator then you will also have to perform the next step

4. In the SQLplus command window type:grant lt_rw to lt;
-

Creating the *LT Analyzer* Database Tables

A *LT Analyzer* database consists of a combination of tables and sequences. To create the database tables follow this procedure.

TASK

1. Log into the desired Oracle instance as the *LT Analyzer* schema owner
 2. In the SQLplus command window copy and paste the text from this file <LT Analyzer Installation Folder>/conf/ORACLEscript.txt and run it, or run the contents of that file as a script.
-

Defining the Privileges for the Roles

To define the privileges for the LT_RW and LT_RO roles follow this procedure.

TASK

1. Open the file <LT Analyzer Installation Folder>/conf/ORACLEroles.txt in a text editor.
2. Edit the file as needed:
 - If you have chosen role names other than the default (default are LT_RW and LT_RO) you will have to update to your names.
 - If you have chosen a schema name other than the default of 'lt' you will have to update to your name.
3. Log into the desired Oracle instance as the DBA (root user).
4. Copy and paste the text from the editor into the SQLplus command window.

ADDITIONAL INFORMATION: If you are creating more than one instance of the *LT Analyzer* schema you will have to add these privileges for each one.

Creating a *LT Analyzer* Administrator in an Oracle Database

Repeat the following steps for all administrator type users of *LT Analyzer*.

TASK

1. Log into the desired Oracle instance as the DBA (root user).
 2. In the SQLplus command window type:

```
create user ltadmin identified by <ltadmin_passwd>
temporary tablespace ltrack
quota unlimited on ltract
grant create session, connect, resource to ltadmin;
grant lt_rw to ltadmin;
```
-

Creating *LT Analyzer* Users in an Oracle Database

Repeat the following steps for all reporting-only type users of *LT Analyzer* follow this procedure.

TASK

1. Log into the desired Oracle instance as the DBA (root user)
 2. In the SQLplus command window type:

```
create user ltuser identified by <ltuser_passwd>
temporary tablespace temp
default tablespace ltrack
quota unlimited on ltrack;
grant create session, connect, resource to ltuser;
grant lt_ro to ltuser;
```
-

Necessary Information for Users

All *LT Analyzer* users (both administrator and reporting-only) will need to be provided with:

- a server name, an instance name, and a port number,
- a schema name, and
- their user ID and password.

Upgrading the Oracle Database from a Previous *LT Analyzer* Database version

To upgrade an earlier version *LT Analyzer* database to the current version, use the following procedure.

TASK

1. Log into the desired Oracle instance as the DBA (root user).
2. To determine the current database version, in the SQLplus command window type:
`SELECT databaseVersion FROM LTVersion;`
3. To upgrade the *LT Analyzer* database, copy and paste the text in this file /LT Analyzer Installation Folder/conf/
ORACLE-UPDATE-[old_version_number]-[old_version_number + 1]-script.txt onto the MySQL command line or run the contents of the file as a script.

STEP RESULT: All of the tables in the *LT Analyzer* database are upgraded to the next version.

4. Repeat the previous step as necessary with increased version numbers until you have upgraded to the latest version
-

Setting Up a New File Database

Prerequisites

File databases can only be created and/or opened on a Windows XP computer system using the 32 bit version of *LT Analyzer*. It is not possible to read or create file databases using the 64 bit version of *LT Analyzer*.

How to set up a File Database

PREREQUISITE

File database can only be created and opened on a Windows XP system.

Use this procedure to set up a file database

TASK

1. Click **File > Create File Database ...** to open the Create a New Database dialog

ADDITIONAL INFORMATION: When the *LT Analyzer* screen first opens, many of the menu items are shadowed and unavailable.

2. Click **Browse...** to open a directory window and enter a filename for your *LT Analyzer* file database. The Microsoft Access Database filename extension `.mdb` is added automatically. Click **Next** to open the next dialog.
3. In the Import Models dialog, click **Browse...**, beside each text field, and select the model filenames for this database. Click **Next** to open the next dialog.

ADDITIONAL INFORMATION: If these data are not yet available, then *LT Analyzer* allows you to leave these text fields blank and continue with the database creation. However, before you can use the *LT Analyzer* features (for example produce group-based and financial reports), you must eventually add these models with the Import feature or edit the empty models in your database using the editors in *LT Analyzer*. For more information, go to [“Importing License Usage Data”](#) on page 52.

ADDITIONAL INFORMATION: If you include a license model or a budget model, *LT Analyzer* will create a default content scope and assign these models to it.

4. In the **Import Session Data** dialog, for each file to be added:
 - a Click **Add**.
 - b Select the log files containing your session data.
 - c Select the format of the file being imported from the pull down menu at the top of the dialog. Your input file type must match your incoming data format or the session data will not be interpreted correctly. If your session data are not yet available, then *LT Analyzer* allows you to leave these text fields blank and continue with the database creation. However, before you can use any *LT Analyzer* features, you must add these files later using the procedure described in “**Importing License Usage Data**” on page 52.
 - d In the Server field, enter the name of the server that this data comes from.
 - e Enter the name of the software vendor that this data comes from in the Vendor field.

STEP RESULT: The filename(s) appear in the window

5. Click **Next** to open the next dialog.
6. In the **Set Data Preferences** dialog, identify the filters for the incoming data. Click **Finish** to open the **Create Database** dialog and start creating your database.

ADDITIONAL INFORMATION: Choose any, all or none of the following filters. If you leave the screen blank, then all data are accepted.

- **Session Duration**

Select this option to specify the range of seconds defining a session length that you want contained in the database. Type in the minimum and maximum time values (in seconds).

- **All Data**

This accepts all the data in the file without imposing time period filtering.

- **Date Range**

You define the date range in two separate text fields using the same procedures. Fill in the Start Date first, and then the End Date. Choose one of the following procedures:

- Double-click on the text field values and type in new ones using the format **YYYY-MM-DD** (i.e., year-month-day), or

- Open the calendar by clicking the square button beside the text field. Select the year and month from small pull-down menus and then click on the actual calendar day. Close the calendar by clicking the square button, in the upper right corner, to display the date in the text fields. If the calendar is shadowed and is not accessible, then you have selected a date when no data exists in your *LT Analyzer* database. At

this point, you can choose another date when data exists, or import more data to cover the time you need for the report.

- **Previous ___ days**

This accepts all the data in the file generated over the previous "n" days.

7. When *LT Analyzer* finishes creating your database, it displays the message **Task Complete** in the **Create Database** dialog. Click **Close** to display a screen of the resulting database information.
-

Managing Usage Data

Prerequisites

File databases can only be created and/or opened on a Windows XP or Windows 7 computer system.

Opening an Existing Server Database

To open an existing *LT Analyzer* database follow this procedure.

TASK

1. Click **File > Connect to Server Database...** to open the Connect to a **LT Analyzer Server Database** dialog.
ADDITIONAL INFORMATION: You can also select a previously opened database from the list at the bottom of the **File** menu. *LT Analyzer* displays the last ten (10) databases opened.
 2. Fill in the *LT Analyzer* Server Database dialog and click **OK** to have *LT Analyzer* display the database information, data sources and session statistics. At this point, the menu items are active and you can start using any of the *LT Analyzer* features.
-

Opening an Existing File Database

To open an existing *LT Analyzer* database follow this procedure.

TASK

1. Click **File > Open File Database...** to open a window, find the appropriate database and select it.
ADDITIONAL INFORMATION: You can also select a previously opened database from the list at the bottom of the **File** menu. *LT Analyzer* displays the last ten (10) databases opened.

2. Click **Open** to have *LT Analyzer* display the database information, data sources and session statistics. At this point, the menu items are active and you can start using any of the *LT Analyzer* features.

ADDITIONAL INFORMATION: Whether or not the **Admin** menu is accessible to you after opening a database depends on the privileges you have on that database. The **Admin** menu is only available to users who have administration privileges on the server or local database.

Importing License Usage Data

PREREQUISITE

- If you do not have an existing *LT Analyzer* database, then you must create one. See one of the following:
 - [“How To Set Up a MySQL Server Database”](#) on page 35
 - [“How to set up an Oracle Server Database”](#) on page 42
 - [“How to set up a File Database”](#) on page 47
- You must be connected to the database.

Use this procedure to import license usage data into the *LT Analyzer* Database. The table below shows the supported log file formats and their corresponding file extensions.

Table 3: LT Analyzer Log File Formats and Extensions

Format	Extension
FLEXlm™ Debug Log	.out, .log
FLEXlm™ Enhanced Debug Log	.enh
FLEX RIF Report	.raw, .rif
Reprise™ Report	.rep
Petrel™ Usage Report	.pet
Petrel™ SLOG Report	.slog
IBM LUM™ Report	.lum

Format	Extension
Sentinel™ Log	.sen
License Tracker™ ASCII Log	.lta
License Tracker™ Binary Log	.ltb

To add data into an existing *LT Analyzer* database, follow this procedure.

TASK

1. Click **Admin > Import Sessions...** to open the panel of the Import Session dialog.
2. In the Import Session Data dialog, click **Add** to open the Add Session Data dialog.
3. Click **Browse...** to select a log file containing your session data. The filename appears in the window.
 - a You **MUST** select the type of file being imported from the **Format** pull down menu. Your input file type must match your incoming data format or the session data will not be interpreted correctly.
 - b Optional step: **Timezone**: The timezone pull-down menu allows you to select the timezone where the data originated. *LT Analyzer* will adjust all times in the log file to the local time of the database.
 - c Optional step: **Vendor**: You may select or create a new Vendor name to associate with this log file. This is used both for associating a Vendor with any new features found in this log file as well as to enable the AutoDelete capability in AutoLoader.
 - d Optional step: **Server**: You may associate a server with a log file. This is used both in the Filter as well as in the AutoDelete capability in AutoLoader.
 - e Repeat this procedure until all of the files for your database are listed in the window. Click Next to move to the Set Data Preferences panel.

4. In the Set Data Preferences dialog, identify the filters for the incoming data. Click **Finish** to open the Create Database dialog and start creating your database.

ADDITIONAL INFORMATION: Choose any, all or none of the following filters. If you leave the screen blank, then all data are accepted.

- **Session Duration:**

Select this option to specify the range of seconds defining a session length that you want contained in the database. Type in the minimum and maximum time values (in seconds).

- **All Data:**

This accepts all the data in the file without imposing time period filtering.

- **Date Range:**

You define the date range in two separate text fields using the same procedures. Fill in the Start Date first, and then the End Date. Choose one of the following procedures:

Double-click on the text field values and type in new ones using the format YYYY-MM-DD (i.e., year-month-day),

or:

Open the calendar by clicking the square button beside the text field. Select the year and month from small pull-down menus and then click on the actual calendar day. Close the calendar by clicking the square button, in the upper right corner, to display the date in the text fields. If the calendar is shadowed and is not accessible, then you have selected a date when no data exists in your *LT Analyzer* database. At this point, you can choose another date when data exists, or import more data to cover the time you need for the report.

- **Previous ___ days:**

This accepts all the data in the file generated over the previous "n" days.

5. When *LT Analyzer* finishes adding the log files to your database, it displays the message Task Complete in the Import Sessions dialog. Click **Close** to display a screen of the resulting database information.

Exporting Information from a Database

To export information from a *LT Analyzer* database follow this procedure.

TASK

1. Click **File > Export** to open a list of export options.

2. Select the type of data files you want to export.

ADDITIONAL INFORMATION: *LT Analyzer* exports all files, except for charts, in simple ASCII format and automatically adds the extension `.csv`. Charts are exported as `.png` files.

- The options **Users Model...**, **Licenses Model...**, and **Budget Model...**, refer to the user model, the license model and the budget model files.
- The option **Sessions...**, refers to session data that are saved as log files.
- The option **Reports/Charts...**, refers to the currently displayed report.

If the **Report/Chart...** option is shadowed and not available, you need to create and display a report or chart.

STEP RESULT: A directory window opens where you identify the location and filename for the exported data.

Editing Session Information

LT Analyzer offers you the following procedures under **Admin** that allow you to modify information in your database:

- Delete Import Source...
- Delete Sessions...

Deleting Sessions

In a *LT Analyzer* database you use the filter dialog to select the data you want to delete. Note that this is a change from Version 2.8 where you selected the data you wanted to keep and let *LT Analyzer* remove the rest.

How to Delete Sessions

TASK

1. Click **Admin > Delete Sessions...** to open the **Describe the data that you would like to delete** dialog.
2. Select one of the tabs in the dialog.
3. Make your data filtering selections in the area below the tabs.
4. Click **OK to delete the selected data.**

STEP RESULT: Sessions matching your data filtering selections will be deleted from the database.

Deleting Import Sources

In an *LT Analyzer* database the **Delete Sessions by Source** dialog is used to remove usage data from particular source.

How to Delete Session Sources

TASK

1. Click **Admin > Delete Import Source** to open the **Delete Sessions by Source** dialog.
 2. Select the data source to delete from the list.
 3. Click **OK**
-

Setting System Preferences

You may set the following system preferences:

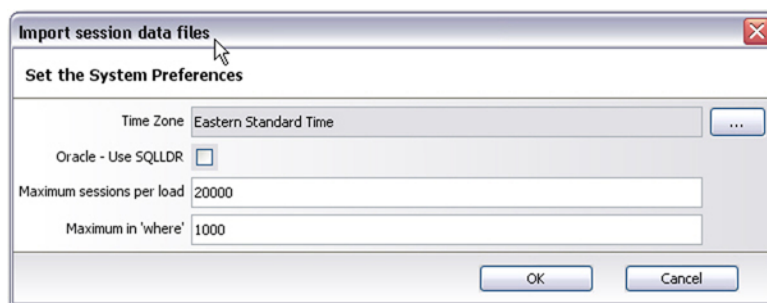
- Time Zone
- Oracle - Use SQLLDR
- Maximum sessions per load
- Maximum in 'where'

How to set system preferences

TASK

1. Click **Admin>System Preferences...** to open the **Set the System Preferences** dialog.

STEP RESULT:



2. In the **Set the System Preferences dialog** enter the required information.
 3. Click OK.
-

Managing Models

LT Analyzer uses four types of models to define how license usage information is analyzed: a user model, a feature model, license models, and budget models. Each one of these models is in turn used to define content scopes and report scopes. as described under “[Managing Scopes](#)” on page 77.

LT Analyzer allows you to edit these model files from inside the application. The following sections describe these procedures.

The Model Manager

LT Analyzer provides a model management and editing interface.

Clicking on these items brings you directly to the appropriate editing dialog.

- **Admin > Manage User Model**
- **Admin > Manage Feature Model**

LT Analyzer uses only one user model and feature model at a time. These contain information on all of the potential users and licenses in the license usage data.

Clicking on the following items brings you first to a model manager dialog containing a list of models. Clicking on a model then brings you to the appropriate editing dialog.

- **Admin > Manage License Models**
- **Admin > Manage Budget Models**

Model Management Dialog

Each of the model management and editing dialogs has the same basic structure and functionality.

When a model manager is first displayed, the dialog shows a list of the existing models available for editing. A button at the bottom of the dialog, **Add...**, allows you to add a new model immediately.

LT Analyzer can use more than one license and budget model that can be applied in analyzing the license usage data.

Selecting one of the existing models in the list activates the other buttons at the bottom of the screen: **Delete**, **Rename**, **Copy**, and **Edit**.

The **Copy** button is used primarily to allow for copying a model, giving it a new name, and then making adjustments to the model.

Double-clicking a model name or pressing the **Edit** button, displays a new dialog designed for editing the selected model. Editing the user, feature, license and budget models is described below. Editing the content and report scopes is described under “**Managing Scopes**” on page 77.

Standard Editing Procedures

This section contains the following topics:

- “**How to prepare a model for editing**” on page 60
- “**How to Edit a Row**” on page 61
- “**How to Edit More than one Row at a Time**” on page 62

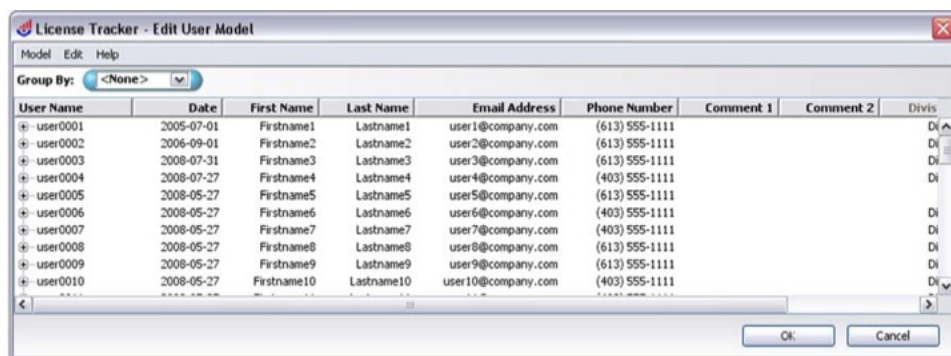
This section describes some general editing procedures that apply to all Models. These procedures assume that you have already opened an editing dialog.

There are two basic procedures for editing model files: direct editing and copying and pasting.

How to prepare a model for editing

To prepare a model for editing follow this procedure

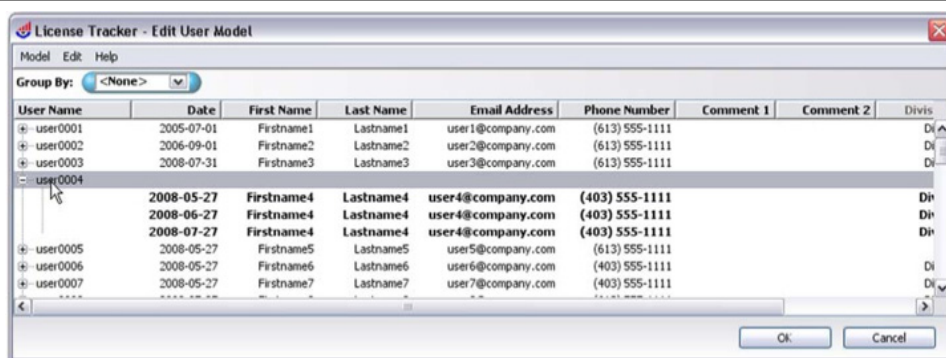
NOTE: When a model is first displayed, the dialog shows the information in collapsed form:



TASK

1. In the **User Model Editing** dialog, click on the + sign in front of the parameter(s) that you intend to edit, or double-click on the parameter(s). This will expand the data into an additional number of so-called "Items", rows of date-related information. Repeat this step till you are seeing the items you need to edit.

STEP RESULT: The display will now look like this:



How to Edit a Row

To edit a row follow this procedure.

TASK

1. Select the desired model name in the **Model Manager** dialog and open the editing dialog by double-clicking the model name or pressing the **Edit** button at the bottom of the **Model Manager** dialog.

STEP RESULT: **Edit Model** dialog is displayed.

NOTE: As the information is initially displayed in collapsed tree form, you may need to expand the item that you are interested in by clicking on the + sign.

2. Click on the + sign in front of the row that you want to edit. Repeat this until you have reached the level (row) that you actually want to edit.
3. Double-click the row that you want to edit. You can also right-click and select **Edit** or select **Edit Menu > Edit**.

STEP RESULT: The **Edit Item** dialog is displayed. This dialog presents data fields appropriate to the model and item you want to edit. Every column in the model has its own field.

4. Enter your new information.

5. Press **OK** to accept your changes.
ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.
STEP RESULT: You return to the **Edit Model** dialog
6. Continue to make changes as desired and then click **OK** or **Cancel**.
ADDITIONAL INFORMATION: Clicking **Cancel** brings you back to the **Model Manager** dialog without saving your changes in the database.
STEP RESULT: Clicking **OK** saves your changes in the database and brings you back to the **Model Manager** dialog.

How to Edit More than one Row at a Time

To edit more than one row at a time, follow this procedure.

TASK

1. Select the desired model name in the **Model Manager** dialog and open the editing dialog by double-clicking the model name or pressing the **Edit** button at the bottom of the **Model Manager** dialog.
ADDITIONAL INFORMATION: As the information is initially displayed in collapsed tree form, you may need to expand the item that you are interested in by clicking on the + sign.
STEP RESULT: The **Edit Model** dialog is displayed.
2. Click on the + sign in front of the row that you want to edit. Repeat this until you have reached the level (rows) that you actually want to edit.
3. Shift-click to select the rows you want to edit then right-click and select **Edit** or select **Edit Menu > Edit**.
STEP RESULT: The **Edit Item** dialog is displayed. This dialog presents data fields appropriate to the model and item you want to edit. Every column in the model has its own field. Some fields will be grayed-out because changes in these fields cannot be applied to more than one item at a time.
4. Select the check box beside a field you want to change for all of the selected rows and change the value in field beside the check box. You can make changes to more than one field at a time.
5. Press **OK** to accept your changes.
ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.
STEP RESULT: You return to the **Edit Model** dialog

6. Continue to make changes as desired and then click **OK** or **Cancel**.

STEP RESULT: Clicking **OK** saves your changes in the database and brings you back to the **Model Manager** dialog. Clicking **Cancel** brings you back to the **Model Manager** dialog without saving your changes in the database.

Working with the User Model

Importing a User Model file

Use the procedures described under “[Standard Editing Procedures](#)” on page 60 to guide you through the general editing of the user model.

This section describes how to import a **user model** file.

TASK

1. Click **Admin > Manage User Model** to open the **Edit User Model** dialog.
 2. Click **Model > Import Users > Import From File**. The **Import User Model** dialog appears.
 3. In the **Valid Date** field, write or select the date from which the user data is valid.
ADDITIONAL INFORMATION: User data changes over time, and it is necessary to track these changes. The Valid Date accomplishes this. The Valid Date is the day after which the user data in the user model can be considered accurate.
 4. In the **User Model** field, enter the path to the user model to import, or click the ... button to the right to open a file browser and select the file manually.
 5. Click **OK**
-

How to Edit the User Model

Use the procedures described under “[Standard Editing Procedures](#)” on page 60 to guide you through the general editing of the user model.

NOTE: To edit the User Model of a LT Analyzer Database, first open the User Model editing dialog in the Model Manager:

Admin > Manage User Model

How to Add a New User

Use this procedure to add a new user to the User Model.

TASK

1. In the open user model, right-click anywhere and select Add User... or select the **Edit > Add User...** menu item.

STEP RESULT: The add User... dialog is displayed.



This dialog presents user data fields. Every column in the User Model has its own field.

2. In the Add User dialog, enter a unique username and other information.
3. Click **OK** to insert the name in the model in alphabetical order.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.

STEP RESULT: You return to the Edit User Model dialog.

How to Edit an Existing User

Use this procedure to edit the information of an existing user.

TASK

1. In the open user model, select the User that you want to edit, and select **Edit > Edit User...** menu item.

STEP RESULT: The Edit User dialog is displayed.

This dialog presents user data fields. Every column in the User Model has its own field.

2. In the Edit User dialog, change the values in the bold fields in the dialog box as required.

ADDITIONAL INFORMATION: The grayed-out fields in the dialog cannot be changed.

3. Click **OK**.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.

STEP RESULT: You return to the Edit User Model dialog.

How to Delete a User

Use this procedure to delete a user from the User Model.

TASK

1. In the open user model, select one or more rows containing the usernames that you want to remove.

2. Click **Edit > Remove User...** or right-click and select **Remove User...** to remove the row of information associated with the username or names.

3. Click **OK** at the bottom of the window to save the change.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.

STEP RESULT: You return to the Edit User Model dialog.

How to Add an Item

Use this procedure to add an item to the User Model.

TASK

1. In the open user model, right-click the User whose Item you want to add, and select **Add Item....**

STEP RESULT: The Add Item dialog is displayed.

This dialog presents user data fields. Every column in the User Model has its own field.

2. In the Add Item dialog change the values in the bold fields in the dialog box as required.

STEP RESULT: The grayed-out fields in the dialog cannot be changed.

3. Click **OK**.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.

STEP RESULT: You return to the Edit User Model dialog.

How to Edit an Existing Item

Use this procedure to edit an existing item in the User Model.

TASK

1. In the open user model, double-click the user whose Item you want to edit, or by clicking on its + sign.

STEP RESULT: One or more Items (rows of date-related user information) are displayed.

2. Double-click the Item that you want to edit. You can also right-click the Item and select **Edit Item...**

STEP RESULT: The Edit Item dialog is displayed.

This dialog presents user data fields. Every column in the User Model has its own field.

3. Change the values in the bold fields in the dialog box as required.

ADDITIONAL INFORMATION: The grayed-out fields in the dialog cannot be changed.

4. Click **OK**.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.

STEP RESULT: You return to the Edit User Model dialog.

How to Delete an Item

Use this procedure to delete an item from the User Model.

TASK

1. In the open user model, double-click the user whose Item you want to delete, and right-click the item in question, and then select **Remove Item**.

STEP RESULT: The Confirm dialog is displayed.

2. This dialog asks for confirmation of the Remove Item request. Click **Yes**.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **No**.

STEP RESULT: You return to the Edit User Model dialog.

How to Add a User Category

Use this procedure to add a User Category to the User Model.

TASK

1. In the open user model, click **Edit > Add Column** or right-click anywhere and select **Add Column**.

STEP RESULT: The New Column Title dialog appears.

2. Enter the name of the user category in the New Column Title dialog.
3. Click **OK**.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.

STEP RESULT: You return to the Edit User Model dialog. A new user category column appears on the right-hand side of the Edit User Model dialog.

How to Add or Edit Attributes Under a User Category

Use this procedure to edit attributes under a User Category.

TASK

1. Follow the procedure described under [“How to Edit a Row”](#) on page 61 or [“How to Edit More than one Row at a Time”](#) on page 62.

ADDITIONAL INFORMATION: You will notice that the Edit User dialog has a field for the various user categories you have defined. Fill in the appropriate field with the appropriate information. You can also define an attribute name for an individual user by following the procedure described under [“How to Edit an Existing User”](#) on page 64.

How to Delete a User Category

Use this procedure to delete a User Category.

TASK

1. In the open user model select any cell in the column of the User Category you want to delete.
2. Click **Edit > Remove Column** or right-click anywhere and select **Remove Column**.

STEP RESULT: A Remove Column confirmation dialog appears.

3. Click **OK** to confirm the deletion.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.

STEP RESULT: You return to the Edit User Model dialog.

Working With LDAP

Connecting to an LDAP Server

PREREQUISITE Familiarity with LDAP systems, terms and administration is strongly recommended.

Before importing user information from LDAP, it is necessary to configure a connection to an LDAP server. To do so, follow this procedure.

TASK

1. Open the **LDAP Configuration Window** by clicking **Admin > LDAP Configuration**.
2. Enter the necessary information in the input fields of the **LDAP Configuration Window**. The Fields are:
 - a **Base DN** (Optional): The Base Distinguished Name of the LDAP tree.
 - b **Object Class** (Required): The LDAP leaf type to search through when finding user information. Defaults to **inetOrgPerson**.
 - c **Security Level** (Required): The security scheme used to login to the LDAP server.
 - d **User DN** (Required): The User Distinguished Name.
 - e **Password** (Required): The password required to logon to the LDAP server.
3. Before continuing, it is necessary to test the connection to the LDAP server. Press the **Test Connection** button to do so.
4. If the connection test failed, verify the information entered in step 2. Otherwise continue.
5. Press the **Next** button in the **LDAP Configuration Window**.
6. In the next window that appears, the correlations between user information in the user model and LDAP system are set. The left hand pane lists the column headings in the user model csv, which are the categories of user information that *LT Analyzer* sees. These can not be changed. On the right are the user attributes in the LDAP system. These can be changed.

ADDITIONAL INFORMATION: If the administrator has defined any custom attributes in the user model, they will appear in the left hand pane.

7. Adjust the attributes of the LDAP system in the right hand pane so that they match up correctly to the attributes in the left hand pane. This process will vary between organization, as organizations often customize their LDAP systems and add new attributes to fit their needs.

ADDITIONAL INFORMATION: Note that the **UserName** attribute of the user model is mapped to the unique user id in the LDAP system, usually **uid**. The **UserName** in *LT Analyzer* must be a unique value, otherwise many functions will fail.

ADDITIONAL INFORMATION: The correlation does not have to be complete in either direction - it is permissible to leave user model attributes with no corresponding LDAP attribute and vice versa. If this is the case for any attribute, then that attribute will simply be left blank in the user model after importing the data.

8. After setting the attributes, click the **OK** button. This completes the procedure.

RESULT: The LDAP system is configured and ready to import user data.

Importing User Data from LDAP

Follow this procedure to import user information from LDAP

TASK

1. Open the **User Model Manager** by clicking **Admin > Manage User Model**.
 2. Click **Model > Import Users > Import From LDAP**
 3. Provide the valid date for the data from LDAP. This is the date from which the user data can be considered accurate, and is used to account for information that changes over time like office location, role, etc. This defaults to today's date.
 4. Click **OK**. The data from LDAP will be imported. This completes the procedure.
-

Editing the Feature Model

Use the procedures described under “**Standard Editing Procedures**” on page 60 to guide you through the general editing of the feature model.

This section describes editing procedures that are specific to the **feature model**.

IMPORTANT: The following procedures assume that you have already opened your feature model in edit mode with **Admin > Manage Feature Model**.

How to Add a New Feature

TASK

1. In an open **feature model** right-click anywhere and select **Add Feature...** or select the **Edit > Add Feature...** menu item.
STEP RESULT: The **Add Feature** dialog is displayed.
 - This dialog presents feature data fields. Every column in the **feature model** has its own field.
2. In the **Add Feature** dialog,
 - a Enter a unique feature name beside **Feature Name:**.
 - b Fill in the fields beside **Vendor Name:**, **Feature Sharing Strategy:** and **False Denial Threshold:** as necessary.
3. Click **OK** to insert the feature in the model in alphabetical order.
ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.
STEP RESULT: You return to the **Edit Feature Model** dialog.

How to Edit an Existing Feature

TASK

1. In the open **feature model** double-click the **Feature** that you want to edit. You can also right-click and select **Edit Feature...** or select the **Edit > Edit Feature...** menu item.
ADDITIONAL INFORMATION: You may want to expand the displayed information by clicking on the + sign in front of the vendor name concerned.
STEP RESULT: The **Edit Feature** dialog is displayed.
 - This dialog presents feature data fields. Every column in the **feature model** has its own field.
2. In the Edit Feature... dialog, change the values in the bold fields in the dialog as required.
3. Click **OK**.
ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.
STEP RESULT: You return to the **Edit Feature Model** dialog.

How to Delete a Feature

TASK

1. In the open **feature model** select one or more rows containing the feature name that you want to remove.

ADDITIONAL INFORMATION: You may want to expand the displayed information by clicking on the + sign in front of the vendor name concerned.

2. Click **Edit > Remove Feature...** or right-click and select **Remove Feature...** to remove the row of information associated with the feature name or names.
3. Click **OK** at the bottom of the dialog to save the change.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.

STEP RESULT: You return to the **Edit Feature Model** dialog.

How to Assign a Feature Alias

PREREQUISITE Before performing this procedure, it is necessary to have installed a feature model in the *LT Analyzer* database.

A Feature Alias is a name for a feature that appears in the *LT Analyzer* interface and is substituted for the feature name found in the log files. To assign a feature alias to a software feature, follow this procedure.

TASK

1. Open the **Feature Model Manager** by clicking **Admin > Manage Feature Model**.
 2. In the **Feature Model Manager**, select a feature in the list and right-click.
 3. In the drop-down menu, select **Edit Feature**. The **Edit Feature Dialog** appears.
 4. In the Feature Alias field, enter the desired value.
 5. Click **OK** in the **Edit Feature Dialog** and click **OK** again in the **Feature Model Manager** to save the changes. This completes the procedure.
-

Working with the License Model

Importing a License Model

License Models must be imported into *LT Analyzer* before they can be used. To so, follow this procedure.

TASK

1. Open **Admin > Manage License Models**
2. Click **Add** in the dialog that appears.
3. Select **Model > Import > Import License Model**
4. Browse to and select the desired License Model csv file and click **OK**
5. Provide the License Model name. This is the name that the License Model will have in *LT Analyzer*.

ADDITIONAL INFORMATION: If this license model will be associated with an account, it is recommended that the name of the license model be similar to the account name.

6. Click **OK** to complete the procedure.
-

Editing a License Model

Use the procedures described under “[Standard Editing Procedures](#)” on page 60 to guide you through the general editing of a license model.

This section describes editing procedures that are specific to a **license model**.

ATTENTION: The following descriptions assume that you have already opened a license model in edit mode with **Edit > License Model**.

How to Add a New License

This procedure consists of adding a new row with a unique license name.

TASK

1. In an open license model click **Edit > Add License...** to open a dialog for entering information. If data are already present in the model, then you can also right-click and select **Add License...**

ADDITIONAL INFORMATION: You may want to expand the displayed information by clicking on the + sign in front of the vendor name concerned.

2. In the **Add License** dialog, select a vendor name and feature name from the pull-down menus at the top of the dialog.
3. Select or type in a date in **the Date** field.
4. Fill in the other fields with the new license information.

ADDITIONAL INFORMATION: Only enter data in those fields that require information for your purposes and leave the other fields blank (with zero values).

ADDITIONAL INFORMATION: You can move between fields with the **TAB** key.

5. Click **OK** at the bottom of the dialog to save the change.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.

STEP RESULT: You return to the **Edit Feature Model** dialog.

How to Edit an Existing License

To edit an existing license in a license model:

TASK

1. In the open **license model** double-click the license that you want to edit. You can also right-click and select **Edit License...** or select the **Edit > Edit License...** menu item.

ADDITIONAL INFORMATION: You may want to expand the displayed information by clicking on the + sign in front of the vendor name concerned.

STEP RESULT: The **Edit License** dialog is displayed.

- This dialog presents license data fields. Every column in the **feature model** has its own field.
2. Change the values in the fields in the dialog as required.
 3. Click **OK**.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.

STEP RESULT: You return to the **Edit License Model** dialog.

How to Delete License Items

To delete one or more licenses from a license model:

TASK

1. In an open **license model** select the license or licenses you want to delete.
ADDITIONAL INFORMATION: You may want to expand the displayed information by clicking on the + sign in front of the vendor name concerned.
2. Click **Edit > Remove License...** or right-click and select **Remove License....** to remove the row of information associated with the feature name or names.
3. Click **OK** at the bottom of the dialog to save the change.
ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.
STEP RESULT: You return to the **Edit License Model** dialog.

Working with the Account Model

Creating an Account Model

PREREQUISITE Before using this procedure, there must be at least one License Model imported in the *LT Analyzer* database.

Account Models do not exist without License Models. Each Account Model, when created, must be tied to a License Model, To create an Account Model, use this procedure.

TASK

1. Open **Admin > Manage Account Models**
 2. Click **Add** in the dialog that appears. The **Edit Account Model** dialog appears.
 3. From the entry field at the top of the window, select a **License Model** with which to associate this **Account Model**.
 4. Optional Step: Enter a model description in the entry field at the top of the window.
 5. Click the **OK** button at the bottom of the window.
 6. Enter an account name in the dialog that appears. The account name is what will be displayed in the interface, i.e. when creating reports.
-

Editing an Account Model

PREREQUISITE In order to edit an **Account Model**, it is necessary to have already created one.

To edit an **Account Model** follow this procedure.

TASK

1. Open **Admin > Manage Account Models**
 2. Select an **Account Model** from the list.
 3. Click the **Edit** button.
 4. In the dialog that appears, enter the new information for this **Account Model**.
 - a In the **License Model** entry field at the top of the window, select a new **License Model** from the drop down menu.
 - b Enter a new description in the Model Description field at the top of the window.
 5. Click **OK** to complete the procedure.
-

Adding Users to an Account Model

There are three ways to add a user to an account model:

- Add users to an account automatically when using AutoLoader. See [“Using Auto-Loader”](#) on page 83 for details.
- Edit the User Model directly and enter the account name into the **Account** column for each user. See [“Working with the User Model”](#) on page 63 for details.
- Associate an LDAP attribute with the Account column in the user model when configuring LDAP. See [“Working With LDAP”](#) on page 68 for details.

Editing a Budget Model

Use the procedures described under [“Standard Editing Procedures”](#) on page 60 to guide you through the general editing of a budget model. This section describes editing procedures that are specific to editing budget models.

ATTENTION: The following descriptions assume that you have already opened a budget model in edit mode with **Edit > Budget Model**.

How to Add a Year to a Budget Model

To add a new budget item:

TASK

1. In an open budget model click **Edit > Add Year...** to open the **Add Year** dialog for entering information.
 - a If data are already present in the model, you can select an existing year and right-click and select **Add Year...** or click **Edit > Add Year...** Values in the selected year are copied into the Add Year dialog automatically.
2. In the **Add Year** dialog, select a year from the pull-down menus at the top of the dialog.

ADDITIONAL INFORMATION: You can move between fields with the **TAB** key.
3. Fill in the other fields with the monthly budget information as required.

ADDITIONAL INFORMATION: Only enter data in those fields that require information for your purposes and leave the other fields blank (with zero values).
4. Click **OK** at the bottom of the dialog to save the change.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.
STEP RESULT: You return to the **Edit Budget Model** dialog.

How to Edit an Existing Budget Model

To edit an existing budget item:

TASK

1. In the open **budget model** double-click the budget year that you want to edit.
 - a You can also select the year and right-click and select **Edit Year...** or
 - b You can also select the year and then select the **Edit > Edit Year...** menu item.

STEP RESULT: The **Edit Year** dialog is displayed.

 - This dialog presents budget year data fields. Every column in the **budget model** has its own field.
2. Change the values in the fields in the dialog as required.
3. Click **OK**.

ADDITIONAL INFORMATION: To leave the dialog without making any changes, press **Cancel**.
STEP RESULT: You return to the **Edit Budget Model** dialog.

Managing Scopes

LT Analyzer uses the concept of "scopes" to help make using *LT Analyzer* easier for end-users.

Administrators can create two types of scopes:

Content Scopes

Allow an administrator to give a name to a view into the license usage data. For example, content scopes called "Office-Houston" and "Office-Calgary" might be used to generate reports on license usage in those locations. Another set of content scopes called "Vendor-1" and "Vendor-2" might be created to generate reports on particular software packages or features, in all locations.

Report Scopes

Allow an administrator to give a name to preselected suite of reports. For example a report scope called "Division-Quarter-End" might be used to generate reports on budgeted versus actual expenditure on software licensing at the divisional level for regular quarterly budget reviews. Another report scope called "Weekly-Feature-22-Div-2" might be used to generate reports on a particular software license in a particular division.

LT Analyzer allows you to edit scopes from inside the application. The following sections describe these procedures.

The Scope Manager

LT Analyzer provides a scope management and editing interface.

Clicking on of these items brings you directly to the appropriate editing dialog.

- **Admin > Manage Report Scopes...**
- **Admin > Manage Content Scopes...**

Clicking one of these menu items brings you directly to a scope manager dialog containing a list of scopes. Clicking on a scope then brings you to the appropriate editing dialog.

Scope Management Dialog

Both scope management dialogs have the same basic structure and functionality.

When a scope manager is first displayed, the dialog shows a list of the existing scopes available for editing. A button at the bottom of the dialog, **Add...**, allows you to add a new scope immediately.

Selecting one of the existing models in the list activates the other buttons at the bottom of the screen: **Delete**, **Rename**, **Copy**, and **Edit**.

The **Copy** button is used primarily to allow for copying a model, giving it a new name, and then making adjustments to the model.

Double-clicking a scope name or pressing the **Edit** button, displays a new dialog designed for editing the selected scope.

Editing a Content Scope

Use the procedures described under “[Standard Editing Procedures](#)” on page 60 to guide you through the general editing of a content scope.

This section describes editing procedures that are specific to a content scope.

NOTE: When defining a content scope you select user and feature attributes to restrict the content (or view into the database). If no attributes are selected then there are no restrictions (i.e. all of the users or features will be part of the content scope).

Adding a New Content Scope

This procedure consists of adding a new row with a unique content scope name.

TASK

1. Click on **Admin > Manage Content Scopes...** to display the **Edit Content Scope** dialog.
2. In the open **Edit Content Scope** dialog, click **Add...** at the bottom of the screen to open a second-level **Edit Content Scope** dialog.
3. In the open **Edit Content Scope** dialog you may type a brief description of the scope in the text field beside Model Description (optional).

4. Click on the **Users** tab below the scope name field.

STEP RESULT: The dialog shows three active areas:

- A pull-down selection box containing a list of user categories (or group types) immediately below the tabs.

NOTE: The categories listed in this box reflect the contents of the User Model. New categories can be created by editing the User Model. See [“How to Edit the User Model”](#) on page 63 for more information.

- A list of attributes for the user categories (or groups) below the selection box.
 - An empty area on the right hand side of the dialog.
5. Select a category from the pull-down selection box to display a list of attributes in that category.
 6. Select the attributes you want included in this scope and use the right-hand arrow to move the user(s) into the empty area on the right-hand side of the dialog. Users having these attributes in this area will be included in the scope.
 7. Repeat the previous two steps until all of the attributes in all of the categories you want are included in the scope are listed on the right hand side of the dialog. Note that if attributes from multiple categories are included that only users with one of the attributes from each of the categories will be included in the content scope.
 8. Click on the **Features** tab below the scope name field.

STEP RESULT: The dialog shows two active areas:

- A list of vendors below the tab.

NOTE: The features listed in this dialog reflect the contents of the Feature Model. New features can be created by editing the feature model.

- An empty area on the right hand side of the dialog.
9. Select the vendors whose features you want included in this scope and use the right-hand arrow to move the vendor(s) into the empty area on the right-hand side of the dialog. Features from the vendors in this area will be included in the scope.

10. Click on the **Data Source** tab.

STEP RESULT: The dialog shows two active areas:

- A list of servers below the tab.

NOTE: The servers in this area represent license servers that the data comes from.

- An empty area on the right hand side of the dialog.

11. Select the servers whose features you want included in this scope and use the right-hand arrow to move the vendor(s) into the into the empty area on the right-hand side of the dialog. Data from the servers in this area will be included in the scope.

12. Click on the **Models** tab below the scope name field.

STEP RESULT: The dialog shows two pull-down selection boxes:

- A **License Model** selection box.
- A **Budget Model** selection box.

NOTE: The models listed in these selection boxes in this dialog reflect the license and budget models associated with the license usage data. New license and budget models can be created. See "Editing a License Model" on page 54 and "Editing a Budget Model" on page 56 for more information.

13. Select the license and budget models you want included in this scope. The selected models will be included in the scope, and used for all financial analyses.
14. Click on **OK** at the bottom of the dialog to display the Content Scope Name dialog.
15. Type the name of the new scope in the text field beside Scope Name.
16. Click on **OK** at the bottom of the dialog.

STEP RESULT: You returned to the list of content scopes. The scope you have created is included in the list.

How to Delete a Content Scope

TASK

1. Click on **Admin > Manage Content Scopes...** to display the **Edit Content Scope** dialog.
 2. In the open **Edit Content Scope** dialog, select a scope you want to delete.
 3. Click **Delete** at the bottom of the screen to delete the scope.
-

Editing a Report Scope

Use the procedures described under “**Standard Editing Procedures**” on page 60 to guide you through the general editing of a report scope.

This section describes editing procedures that are specific to a report scope.

How to Add a New Report Scope

This procedure consists of adding a new row with a unique report scope name.

TASK

1. Click on **Admin > Manage Report Scopes...** to display the **Edit Report Scopes** dialog.
2. In the open **Edit Report Scopes** dialog, click **Add...** at the bottom of the screen to open a second-level **Edit Report Scope []** dialog.
3. In the open **Edit Report Scope []** dialog you may type a brief description of the new scope in the text field beside **Description** (optional).
4. Type the name of the new scope in the text field beside **Scope Name**.
5. Click on a tab below the scope name to display a list of available reports.
6. Select a report to be included in the scope by clicking the checkbox beside a report name. Reports with green checkmarks beside the name will be included in the scope.
7. Repeat the previous two steps until all of the reports you want in the scope have been selected.
8. Click on **OK** at the bottom of the dialog to display the **Content Report Name** dialog.
9. Type the name of the new scope in the text field beside **Scope Name**.
10. Click on **OK** at the bottom of the dialog.

STEP RESULT: You returned to the list of report scopes. The scope you have created is included in the list.

How to Edit a Report Scope

TASK

1. Click on **Admin > Manage Report Scopes...** to display the **Edit Report Scope** dialog.
2. In the open **Edit Report Scope** dialog, select a scope you want to edit.

3. Click **Edit...** at the bottom of the screen or double-click a scope name to open a second-level **Edit Report Scope [scope name]** dialog.
 4. Follow the procedures already described under **“How to Add a New Report Scope”** on page 81 to modify the scope.
-

How to Delete a Report Scope

TASK

1. Click on **Admin > Manage Report Scopes...** to display the **Edit Report Scope** dialog.
 2. In the open **Edit Report Scope** dialog, select a scope you want to delete.
 3. Click **Delete** at the bottom of the screen to delete the scope.
-

Automating *LT Analyzer*: AutoTracker

LT Analyzer has a built-in capability for automating certain functions through Auto Tracker. These functions include:

- loading data with **AutoLoader**,
- generating Web Tracker reports with **AutoReporter**, and
- generating and sending alerts by e-mail with **AutoAlerter**

Each of these functions is controlled by a batch process that can be set up by an administrator. An individual batch or script file can contain many lines, each one controlling an individual invocation of one of the **AutoTracker** functions.

AutoTracker Database Specification

Fundamental to running any of the AutoTracker functions is an AutoTracker database specification file.

For a server database, this database specification file contains information that allows the AutoTracker functions to connect to the database. Creation of an AutoTracker database configuration file is done through the dialog accessed via **File > Create AutoTracker DB Configuration File...** These files are saved in the *LT Analyzer*conf directory with a .dbconf file extension.

For a file database the database specification file contains the full path to the MS-Access database, also known as an .mdb file.

Using AutoLoader

AutoLoader enables you to batch schedule the loading of usage log files and models into a *LT Analyzer* database. Each invocation of AutoLoader will add the contents of a specified directory into a specified database. A single script file can contain many lines, each loading the data from a different directory into a *LT Analyzer* database, or even into different databases.

AutoLoader uses the file extension of the log being imported to determine the log format. AutoLoader only uses the extension to determine the format, so the log file extension must be accurate, and must match the format. There is no option that allows the log format to be specified separately from the extension.

The table below shows the log formats that are supported by AutoLoader and their corresponding extensions.

Table 4: AutoLoader Log File Formats and Extensions

Format	Extension
FLEXlm™ Debug Log	.out
FLEX RAW Report	.raw
FLEX RIF Report	.rif
Petrel™ SLOG Report	.pet
Sentinel™ Log	.sen
License Tracker™ ASCII Log	.lta
License Tracker™ Binary Log	.ltb

Command Line Syntax

The command line syntax used when invoking AutoLoader or writing scripts is as follows:

```
<preliminaries/context> <required items> [options]
```

where:

- `<preliminaries/context>`
Are the basic commands required to run AutoLoader.
- `<required items>`
is the minimum set of commands required for AutoLoader to load data.
- `[options]`
are variables that allow the administrator to select and fine tune the data that will be loaded.

NOTE: AutoLoader must be run from the 'bin' directory of your LT Analyzer installation.

Preliminaries and Context for the Script

The Preliminaries and Context for AutoLoader scripts are basic commands that set up the Java Virtual Machine to run the AutoLoader software. They are:

- The Java Virtual Machine invocation. On Windows, this invocation takes the form:

```
c:\windows\system32\javaw.exe
```

On Solaris and Linux, the invocation is:

```
java
```

- Some Java Virtual Machine options:

```
-Xmx1024m
```

- The Classpath specification:

```
ChartDirector_s.jar;licensetracker.jar;activation.jar;mail.jar;mysql-connector-java-5.0.4-bin.jar;ojdbc14.jar
```

NOTE: The separator for Windows is ";" and for Solaris it is ":"

- Finally, the Main class:

```
com.licensetracker.autoloader.AutoLoader
```

Required Items

After preparing the Java Virtual Machine, some information must be sent to AutoLoader itself. These are the required items:

Database specification

```
-db <dbspec>
```

the name of the dbconf file (without the extension) to connect to server database, or full path to the file database.

Input directory

```
-i <input_directory>
```

The directory where AutoLoader will find input files. The input directory can be on the local machine or on a remote server. Typically this directory is populated by another periodic process that collects license usage logs from other servers.

NOTE: The input directory must exist before the script is run.

Default server name

-s <serverName>

the name of the license server to be associated with this import session; if a server name is found in the file it will override this value

Default vendor name

-v <vendorName>

the Vendor name to be associated with any new features encountered in this session import

Options

Options are passed to AutoLoader to change its behaviour or settings. Options are not required to run AutoLoader. The options are:

- Restricting users and/or features to be imported

-[u | f | uf] <scope>

Whether to restrict users, features or both to those specified by the content scope named "scope".

NOTE: If the scope is not specified, the entire database will be the scope.

- Time offsetting for time zone compensation

-t <nHours>

The number of hours that times read in are to be adjusted by to correspond to the time zone of the database.

- Date range: startDate and endDate

-sd <date>

Only data after this date will be loaded. The required date format is "MMM dd, yyyy HH:mm:ss".

NOTE: If the start date is not specified, everything from the beginning of the database will be used.

-ed <date>

Only data before this date will be loaded. The required date format is "MMM dd, yyyy HH:mm:ss"

NOTE: If the end date is not specified, everything to the end of the database will be used.

- Date range: a specific number of previous days or previous months
`-pd <count>`
Only data from the previous number of days specified by "count" will be loaded.
`-pm <count>`
Only data from the previous number of months specified by "count" will be loaded.
- Auto-deleting a previous file
`-ad`
Prior to loading the file, AutoLoader will delete session data associated with a previous file which has the same filename, start date, server, and vendor. This enables daily updating while still capturing long session information.
- Re-sync data from LDAP
`-ldap`
Re-sync user data from LDAP before loading the usage data. Note that *LT Analyzer* must be connected to an LDAP system in order for this option to function properly.
- Assign new users to account
`-act <account name>`
If any user is not found in the User Model but is found in the usage data, that user is automatically assigned to the specified account. If this option is not provided then new users are not assigned to any account.

NOTE:

- Only use one date specification option; if none are used entire database period will be considered.
- Use quotation marks if an option contains a space.

Notes

- 1) **AutoLoader** must be run from the 'bin' directory of your *LT Analyzer* installation.
- 2) Only use one date specification option; if none are used entire database period will be considered.
- 3) Use quotation marks if an option contains a space.

Examples

The following scripts are typical examples of how AutoLoader is used to automate the collection of license usage data.

Example: Load all the data in \input

```
C:\WINDOWS\system32\javaw.exe -Xmx1024m -classpath
ChartDirector_s.jar;licensetracker.jar;activation.jar;mail.jar;mysql-c
onnector-java-5.0.4-bin.jar;ojdbc14.jar
com.licensetracker.autoloader.AutoLoader MySQLlocal c:\AutoLoader\input
```

Example: Load all the data in "\input" from the previous 7 days

```
C:\WINDOWS\system32\javaw.exe -Xmx1024m -classpath
ChartDirector_s.jar;licensetracker.jar;activation.jar;mail.jar;mysql-c
onnector-java-5.0.4-bin.jar ojdbc14.jar
com.licensetracker.autoloader.AutoLoader c:\temp\VendorXQ1.mdb c:\
AutoLoader\input -pd 7
```

Example: Load all the data in "\input" between the dates specified

```
C:\WINDOWS\system32\javaw.exe -Xmx1024m -classpath
ChartDirector_s.jar;licensetracker.jar;activation.jar;mail.jar;mysql-c
onnector-java-5.0.4-bin.jar ojdbc14.jar
com.licensetracker.autoloader.AutoLoader MySQLlocal c:\
AutoLoader\input -sd "Jan 01, 2007 00:00:00" -ed "Mar 31, 2007 23:59:59"
```

Example: Load all the data in "\input" from the last three months

```
java -Xmx1024m -classpath
ChartDirector_s.jar;licensetracker.jar;activation.jar;mail.jar;mysql-c
onnector-java-5.0.4-bin.jar ojdbc14.jar
com.licensetracker.autoloader.AutoLoader OracleAdmin /home/
AutoLoader/input -cs NAUnit -pm 3
```

Using AutoReporter

AutoReporter enables you to batch schedule the generation of *WebTracker* reports. Each invocation of AutoReporter will generate the reports specified in a report scope. A single script file can contain many lines, each generating a different set of reports.

Command Line Syntax

The command-line syntax used when writing AutoReporter scripts is as follows:

```
<preliminaries/context> <required items> [options]
```

where:

- `<preliminaries/context>`
are the basic commands required to run AutoReporter.
- `<required items>`
is the minimum set of commands required for AutoReporter to analyze data.
- `[options]`
are variables that allow the administrator to fine tune the output.

NOTE: AutoReporter must be run from the 'bin' directory of your LT Analyzer installation.

Preliminaries and Context for the Script

The Preliminaries and Context for AutoReporter scripts are basic commands that set up the Java Virtual Machine to run the AutoReporter software. They are:

- The Java Virtual Machine invocation. On Windows, this invocation takes the form:

```
c:\windows\system32\javaw.exe
```

On Solaris and Linux, the invocation is:

```
java
```

- Some Java Virtual Machine options:

```
-Xmx1024m
```

- The Classpath specification:

```
ChartDirector_s.jar;licensetracker.jar;activation.jar;mail.jar;mysql-connector-java-5.0.4-bin.jar;ojdbc14.jar
```

NOTE: The separator for Windows is a semicolon ";" and for Solaris it is a colon ":"

- Finally, the Main class:

```
com.licensetracker.autoreporter.AutoReporter
```

Required Items

After preparing the Java Virtual Machine, some information must be sent to AutoReporter itself. These are the required items are:

- `-db <dbspec>`
the name of the dbconf file (without the extension) to connect to server database, or full path to the file database.
- `-o <output directory>`
The directory in which AutoReporter will put Web Tracker output. The output directory can be on the local machine or on a remote server.
NOTE: The output directory must exist before the script is run.

Options

Options are passed to AutoReporter to change its behaviour or settings. Options are not required to run AutoReporter. The options are:

Scopes:

- `-cs <scope>`
The content scope. This option limits the portion of the database that will be used. If the content scope is not specified the entire database will be used.
- `-rs <scope>`
The report scope. This option specifies the reports and components to use. If the report scope is not specified, all reports with all components and long session threshold of 24 hours in the database will be used.
- `-act <account name>`
This option specifies which account to consider for the reports. Only data from users in this account will be included. This option overrides the `-cs` option.

Date range: startDate and endDate

- `-sd <date>`
Only data after this date will be loaded. The required date format is "MMM dd, yyyy HH:mm:ss".
NOTE: If the start date is not specified, everything from the beginning of the database will be used.
- `-ed <date>`
Only data before this date will be loaded. The required date format is "MMM dd, yyyy HH:mm:ss"

NOTE: If the end date is not specified, everything to the end of the database will be used.

Date range: specific number of previous days or previous months

- `-pd <count>`

Only data from the previous number of days specified by "count" will be loaded.

- `-pm <count>`

Only data from the previous number of months specified by "count" will be loaded.

Indices: multiple hierarchical report sets

- `-i1 <type>:< category[:label]>`

The primary index.

`type` is User or Feature.

`category` is any user category or Vendor.

`label` is the word to use in the Web Tracker navigation header, if desired.

- `-i2 <type>:< category>[:label]`

The secondary index, if desired.

NOTE: The primary index must be defined first.

NOTE:

- Only use one date specification option; if none are used entire database period will be considered.
- Use quotation marks if an option contains a space.

Notes

- 1) **AutoReporter** must be run from the 'bin' directory of your *LT Analyzer* installation.
- 2) Only use one date specification option; if none are used entire database period will be considered.
- 3) Use quotation marks if an option contains a space.

Using the AutoAlerter

AutoAlerter allows an administrator to send alerts to designated users automatically when certain conditions occur in the license usage data.

Command Line Syntax

The command-line syntax used when writing AutoAlerter scripts is as follows:

```
<preliminaries/context> <required items> [options]
```

where

<preliminaries/context>

are the basic commands required to run AutoAlerter.

<required items>

is the minimum set of commands required for AutoAlerter to generate e-mail alerts.

[options]

is the variable that allow the administrator to run AutoAlerter in debug mode.

Preliminaries and Context for the Script

The Preliminaries and Context for AutoAlerter scripts are basic commands that set up the Java Virtual Machine to run the AutoAlerter software. They are:

- The Java Virtual Machine invocation. On Windows, this invocation takes the form:

```
c:\windows\system32\javaw.exe
```

On Solaris and Linux, the invocation is:

```
java
```

- Some Java Virtual Machine options:

```
-Xmx1024m
```

- The Classpath specification:

```
ChartDirector_s.jar;licensetracker.jar;activation.jar;mail.jar;mysql-connector-java-5.0.4-bin.jar;ojdbc14.jar
```

NOTE: The separator for Windows is ";" and for Solaris it is ":"

- Finally, the Main class:
`com.licensetracker.autoalerter.AutoAlerter`

Required Items

After preparing the Java Virtual Machine, some information must be sent to AutoReporter itself. These are the required items:

- `-db <dbspec>`
the name of the dbconf file (without the extension) to connect to server database, or full path to the file database.
- `-i inputDir`
The directory in which AutoAlerter will find the required AlertModel and Subscription-Model CSV files.
NOTE: The input directory, with AlertModel and SubscriptionModel CSV files, must exist before the script is run.
- `-eh emailHost`
Name of the e-mail service. On Windows systems this can be found in
Outlook > Tools > EmailAccounts > Outgoing Mail Server
- `-ef emailFrom`
Address e-mails will be addressed from.
- `-er emailReturn`
Return address for e-mails.

Options

Options are passed to AutoAlerter to change its behaviour or settings. Options are not required to run AutoAlerter. The options are:

- `-debug log fileName`
An optional debug file. All output will be generated in the debug log file if the `-debug` option is specified.

Input Directory

The input directory must have two CSV files:

Alert Model

The **Alert Model** file must follow this naming convention: `xxxxxxxAM.csv`, where `xxxxxxx` can be any string valid in a filename.

Subscription Model

The **Subscription Model** file must follow this naming convention: `xxxxxxxSM.csv`, where `xxxxxxx` can be any string valid in a filename.

AlertModel CSV file

The **AlertModel** CSV file contains the definitions for the alerts.

There are currently two types of alerts, **Long Session Alerts** and **Denial Alerts**. Each row in the **Alert Model** file contains the parameters for one of these alerts. An alert will be evaluated only if someone is subscribed to the alert in the **Subscription Model** file.

Long Session Alerts Syntax

A row designating a **Long Session Alert** in the `xxxxxxxAM.csv` file must follow the format in the table below. The `AM.csv` may have any number of **Long Session Alert** entries.

Table 5: Long Session Alert Entry in the `AM.csv` file

Column Number	Name	Description
1	LongSessionAlert	Type of alert. "LongSessionAlert" is case sensitive.
2	Name	Name of the alert. This will be used in the subscription model as well as the e-mails sent to the subscribers.
3	Feature	Feature to which this alert applies.
4	Duration	Duration of a session that is considered to be a long session (hours)

Column Number	Name	Description
5	Previous hours	How many hours ago to look for long sessions (in hours). For example, a value of 6 means that any long session within the past 6 hours will generate an alert. A value of 1 means that any long session within the past hour will generate an alert. A value of 0 means that any long session that was active at the last time in the database will generate an alert.
6	MaximumConcurrentPercent	The percent of all available licenses that must be exceeded before an alert is generated. For example, a value of 80 means that 80% of all licenses must be in use when a long session is encountered before an alert is generated. If a long session is detected, but the number of concurrent sessions is only at 20%, then no alert is generated. If a long session is detected, and the number of concurrent sessions is over 80%, then an alert is generated. A value of zero will generate an alert for all long sessions. A value of 100 will suppress long session alerts.
7	emailUsersFlag	Automatically e-mail the users that generated a long session alert. This generates a "polite" e-mail reminding them that long sessions are not making good use of corporate resources. A value of "TRUE" means e-mail the users of long session alerts. A value of "FALSE" means do not e-mail the users.

Denial Alert

A row designating a denial alert in the `xxxxxxxxAM.csv` file must follow the format in the table below. The `AM.csv` may have any number of Denial Alert entries.

Table 6: Denial Alert Entry in the `AM.csv` file

Column Number	Name	Description
1	Denial Alert	Type of alert. "Denial Alert" is case sensitive.
2	Name	Name of the alert. This will be used in the subscription model as well as the e-mails sent to the subscribers.
3	Feature	Feature to which this alert applies.

Subscription Model

The **Subscription Model** CSV file defines which users have subscribed to which alert conditions.

NOTE: At least one person must be subscribed to the alert in order for the alert to be evaluated. If no-one is subscribed, then the alert will not be run.

A user can be subscribed to more than one alert. When **AutoAlerter** is run and more than one alert condition is detected, and the user has subscribe to many alerts, then only a single e-mail is sent to the user outlining all of the alert conditions that were detected.

Many users can subscribe to each alert. When **AutoAlerter** detects the alert condition, then an e-mail is sent to all subscribing users.

The subscription model file contains two columns:

- the mail address for the user who is subscribing to this alert
- the name of the alert. The alert name must correspond exactly to one listed in the second column in the **Alert Model** file.

Table 7: A Subscription Model File

Email Address	Alert Name
person@CompanyB.com	Lic22_Campers
person@CompanyB.com	Lic22_Denials
person1@Dept_001.CompanyA.com	SoftwareFeatureX Long Sessions
person2@Dept_999.CompanyA.com	SoftwareFeatureX Long Sessions

NOTE: The first row in the subscription model is for titles and is ignored.

Support and Contact Information

Support Services

All customers who purchase *LT Analyzer* receive access to the User and Admin Manuals that ship with *LT Analyzer*. The manuals can be found in the `doc/` folder of the *LT Analyzer* installation.

License Tracker Inc provides premium technical support services, user forums and webinars to customers with a support contract. If you have a support contract, please go to the **LT-PremiumAssist** portal at www.licensetracker.ca/support.

Customers cannot access the premium support service without a contract. If you would like to purchase a support contract, please go to www.licensetracker.ca/support/purchase.asp

Contact Information

License Tracker Inc can be reached by the following methods:

On the Web

<http://www.licensetracker.ca>

General e-mail

sales@licensetracker.ca

Support e-mail

support@licensetracker.ca

Page left blank on purpose

