SuperLuminate

version 1.9.0.120

User
Guide

-- This page was intentionally left blank –

Table of Contents

What is SuperLuminate..5
- SuperLuminate 5
- Data Dictionary (short definition)..5
- Metadata (short definition)...5
- Secure and Flexible Administration 7
- Powerful Data Analytics 8
- SuperLuminate Based on Modern Industry Standards..8

What is a Data Dictionary...9
- Data Dictionary...10
- Data Dictionary Definition References..15

User Quick Start (online demo)..17
- SuperLuminate (from SuperLuminate help menu)...17
  - Notation Convention...17
  - Basic Look and Feel 17
  - Selection by Taxonomy 18
  - Selection by Entry 18
  - Compound Selection 19
  - Following the Parent Child Relationship Links 19

Navigation Architecture...21
- Notation..21
- Page Header Menu..21
- Logon Page..24
- Main Menu...25
- Search Page (search example #1)...26
  - Search Results Page (search example #1)...27
  - Selection Display Page (search example #1)..28
- Search Page (search example #2)...29
  - Search Results Page (search example #2)...30
  - Selection Display Page (search example #2)..31
- Generalized Reporting...33
- SuperLuminate Lite (Glossary Read Only Access)..34
  - SuperLuminate Lite (glossary) selection page:...36
  - SuperLuminate Lite (glossary) display page:...37
- Administration Menu...38
- Modify Configuration Page...39
- Manage Schema Menu..41
- Chang to User Guide 42
- Flat File Adapters..42
- This page can be used by your SuperLuminate administrator to backup the database and restore the entire data dictionary if required. The administrator can also import and export data from and to comma delimited text files (CSV)..42
- Backup Restore Adapters...44
- Database Adapters...44
- Page Hierarchy..48

Application Architecture..49
  Security..49
  Search and Retrieval...51
  Global Variables...53
Functional Architecture...55
  Functional Segmentation..55
  Function Calling Protocol...55
Conceptual Architecture...60
  Three Schema Architecture...60
  Conceptual Object Design...61
  Schema Modification (A.K.A. Information Model)..63
    Taxonomy...63
    Entity Relationship Diagrams (ERD)...63
    Out-of-the-box (default) Schema and Taxonomy..63
    Taxonomy to Schema Interaction..70
  Data Abstraction 72
    Data Abstraction Level (0)...73
    Data Abstraction Level (1)...74
    Data Abstraction Level (2)...75
    Data Abstraction Level (3)...76
Technical Architecture..77
  Database Naming Conventions..77
  Versioning..78
  Copying..79
  Logical Model (not including auxiliary tables)...80
  Physical Table Definitions..81
    Metaschema Tables...81
    Schema Tables...83
    Auxiliary Tables..84
Software Specifications...86
  SuperLuminate Folders and Application Software..86
Administration Topics...88
  User Maintenance (adding or modifying a user)..88
  Schema Management..88
  System Configuration...88
  Database Backup and Recovery 91
  System Monitoring...92
  External Metadata Importing and Exporting..92
  End of User Guide...94

What is SuperLuminate

SuperLuminate

SuperLuminate (The open source business data dictionary) is a fully functional, web based, thin client, open source data dictionary application developed and distributed by SuperLuminate (The Company).

A data dictionary is important to any organization. The metadata in your data dictionary plays a key role in reacting quickly to new opportunities, reducing costs, and remaining competitive.

SuperLuminate Provides Real-Time Metadata Management

- Easy to Learn Interface
- Intuitive Navigation
- Canned and Open SQL reporting interface

Data Dictionary (short definition)

A data dictionary is a tool for the collection and maintenance of business and technical definitions that are classified and cataloged by subject and then related to other definitions for ease of maintenance and retrieval.

Equally, the data dictionary maintains information about the definitions, including:

- Security information to track creator and modifier
- Version information to track changes to the definition
- Relationship information to understand the description's association with other descriptions of the same or other types.

Metadata (short definition)

In a data dictionary definitions are commonly known as "metadata." Metadata is usually defined as "data about data" because metadata is itself data and is used to describe other data or process.

Metadata is generally broken into two categories, Business metadata and Technical metadata. Business metadata defines your common business vocabulary – your policies, procedures, and rules. Technical metadata defines your physical data assets, your information systems infrastructure – your servers, databases, and reports.

Information Model

The structure of the metadata stored in SuperLuminate is defined by the Information Model. The Information Model in SuperLuminate encompass both the taxonomy and schemas (A.K.A. Information Models).

Taxonomy

The taxonomy enables the SuperLuminate administrator to define a high level standard system of classification for the organization. The SuperLuminate taxonomy comprises a four level hierarchy from category to subject to class and finally to type. The four tier taxonomy was designed specifically to categorize Data Administration metadata.

For example in biological sciences the taxonomy for classifying organisms is based mainly on physical similarities and comprises a hierarchy of seven primary levels.

They are from top to bottom:

- *Kingdom*
- *Phylum*
- *Class*
- *Order*
- *Family*
- *Genus*
- *Species*

Schema (A.K.A. Information Model)

A schema defines the metadata in terms of object types and their relationships. The schema is thus the language for describing the metadata the repository will store.

SuperLuminate provides a default "out-of-the-box" taxonomy and schema. Users can define their own or modify the one provided to meet their specific metadata needs.

NOTE: The highest level in the taxonomy (Category) serves as a high level partition – a way to separate large groupings of information. (Subject) is synonymous with schema name. Schema is "Too Tech" to have on the front-end so we called it Subject. The Subject name "Application" is the "Application" schema; all of the Class names

(Classes) within the "Application" Subject form the entities in the "Application" schema. In this way SuperLuminate can be used to manage multiple schemas.

Extensibility

Extensibility is an important feature that enables users to add information to SuperLuminate for topics specific to their organization. The SuperLuminate administrator can extend the Information. If your organization needs to store information that is not part of the base Information Model, you can extend the model to add new objects and corresponding relationships. SuperLuminate administrators can also add new properties to track information for existing objects.

Versioning

Versioning is an important feature of SuperLuminate. With versioning, as updates happen to objects in the data dictionary. The changes are captured, and the data dictionary maintains a history of the changes.

Metadata Interchange

SuperLuminate is built on a relational database, and uses an open industry standard physical metaschema (IRDS) based on unified modeling language (UML). For these reasons, users have the ability to share metadata across multiple tools from multiple vendors.

Secure and Flexible Administration

Because certain metadata may be sensitive in your organization, you must be cautious how it is distributed. Even if it is relevant to a department, it may not be in the best interest of the enterprise to make certain metadata available to all who are interested in it. Security in SuperLuminate is maintained at the individual user and group levels.

- Web Encryption via Open_SSL
- User Authentication – login User ID and Password
- User Authorization – Administrator, User, Group for Record Access and Maintenance, Reporting, and Tool use
- User Profile Management

Powerful Data Analytics

- Browse All Application Metadata in Real-Time - Absolute confidence that you are looking at the current version of the truth
- Advanced Search Capabilities - Speed investigation of application and data issues with detailed and flexible search facilities
- Data Lineage Analysis - Understand the source and destination of data as it flows through your applications
- Dependency Analysis (Impact Analysis) - Track the impact of systems changes across the enterprise
- Mapping View Analysis - Uncover how data is transformed as it moves between systems
- Reporting - Create and save custom reports for easy repeat access to frequently used data sets
- Versioning - Track changes to your business rules and corporate standards

SuperLuminate Based on Modern Industry Standards

- SuperLuminate is compliant with the LAMP (Web Based Thin-Client) architectural standard
- SuperLuminate is metaschema compliant with IRDS standards
- SuperLuminate is compliant with XML metadata interchange based import and export standards
- SuperLuminate has multi-language Support
- SuperLuminate can be run on any operating system supported by PHP
- SuperLuminate can be run on any relational database supported by PHP

SuperLuminate is based on an industry standard physical architecture (IRDS) that enables your company to extend the functionality of SuperLuminate by modifying the out-of-box taxonomy and schema (A.K.A. Information Model), or by adding one or more of your own schemas. All extensions to SuperLuminate can be accomplished by modifications to the SuperLuminate control data forgoing any need to physically change the application or physical database.

Using a tag line for SalesForce.com, No Software – A SaleForce.com registered trademark, all extensions to SuperLuminate (your company's specific requirements and standards) can be implemented by your data dictionary administrator by way of internal configuration settings alone.

With SuperLuminate "no coding is required."

Because SuperLuminate does not require physical extensions, your company can easily upgrade to new versions of SuperLuminate as they are released.

What is a Data Dictionary

Editors Note:

The following extended definition of "what a data dictionary is" will seem repetitious to those who already "Grok"(1) what a data dictionary is and why a data dictionary is useful. The basic concept of what a data dictionary is and why every organization needs one is not simple to begin with but becomes clear as you start to pick it up. The repetitious nature of this chapter is intended to help drill home the concepts.

(1)To grok is to share the same reality or line of thinking with another physical or conceptual entity.

Robert A. Heinlein, coined the term in his best-selling 1961 book "Stranger in a Strange Land"

Metadata has become one of the hottest arenas in information technology today. Corporations have realized the value of metadata and the absolute need for it in order for their business to thrive in markets that are becoming more and more competitive.

David Marco, Cutter Consortium
Author of: "Building and Managing the Metadata Repository"

One of enterprises' most-common concerns (and frustrations) is the lack of viable solutions to assist them with maintaining the consistency and gaining an understanding of the metadata stored in multiple formats and locations. Fewer than 30 percent of enterprises have implemented a metadata management solution that addresses this issue, and those that do generally only address a portion of the metadata.

Gartner Group

Data Dictionary

A data dictionary is a collection of data definitions categorized by subject. Equally, data dictionaries provide indications about the data's life cycle, indicating, by whom, when and how the data was created, modified or deleted.

A data dictionary is an organized, formal description of data files that describes physical file attributes, such as record lengths and file types, as well as logical file attributes, such as column names and display formats.

A data dictionary is a vehicle for specifying data collection standards. Insofar as a data dictionary records such standards, it is a useful and necessary tool for enabling the collection of Sarbanes Oxley (SOX) requirements, and essentially describes the meaning of the information to be collected.

Data dictionaries are comprised of object and attribute definitions that include:

- descriptions – what is it
- context – who uses it and why
- data domains – what is the range of possible values
- guide for use – which one of the possible values should I use

When Used by a Database Administrator

A data dictionary is a repository for information about data like a database catalog that stores information about the database such as table names, column names, data types, but is capable of storing information about a much larger range of topics, that includes table descriptions, column descriptions, data type descriptions, along with how they are used, backed up and who in the organization is impacted if an issue occurs.

When Used by a Data Administrator

A data dictionary is a repository to store information about data design, and business processes, e.g. entity and attribute descriptions, business rules, and the relationships that entities have to other entities.

When Used by a Business Manager

A data dictionary is a repository to store information about business related items, e.g. glossary terms, reports, processes, and Key Performance Indicators (KPI).

Metadata

The assets stored in a data dictionary are called metadata -- data about data -- and describe critical factors about your systems and applications, such as where a particular data source is located and the types of data that are used by the system or application. Metadata plays a key role in reacting quickly to new technologies, and is required to remain competitive.

Metadata is the set of data that describes data sources and locations, application systems, data formats, and a dictionary like description for each element. Descriptions can be extensive or simple.

Simple Example:
- Name:
  - Manufacturing Product Number
- Definition:
  - A Manufacturing Product Number represents the unique identifier for products produced by a manufacturer
- Data Type:
  - Alphanumeric
- Size:
  - 18 Characters
- Format:
  - AAA-NN-A-NNNNN-NNN

Metadata can sometimes be described reasonably as data descriptions that tells us about the data we use.

The demands of the business user are changing the nature of metadata. Its previous role of describing and cataloging the multitude of relational database tables in the data warehouse is expanding to include providing information to users about the content, meaning, accuracy and quality of the data they are using to make business decisions.

Business Metadata

Business metadata is data that describes information assets in business terms. Business metadata is stored in the data dictionary and accessed by users to find and understand the information they need. For example, business metadata for a report would contain a description (in business terms) of what the report does and what calculations it contains.

Business metadata plays a critical role in Information Systems because it connects the business user with the relevant data in the enterprise. Business metadata supports the business users' perspective of Information Systems, by using common business terms and providing information about the data in terms of context, understandability, search-ability and usability, rather than in terms of infrastructure and database technology. Business metadata educates users about the origins of information and ensures that they apply the information correctly by describing the rules that govern data use and by identifying valid ways in which data can be leveraged and combined with other data. Without business metadata, business users would have to rely on ad hoc inquiries to Information Systems staff to provide the data definitions and explain the logic behind database tables in order to conduct their analyses.

Business metadata can play a role in inspiring user confidence in the completeness of the data. Business metadata puts the answer in textual, non-technical terms that the business user can understand. It describes the source systems, files and any transformation logic.

Examples of business metadata include: who maintains the data, what is the confidence level of the data and its quality, what algorithm is used to create the values, what is the definition of the data, and what reports are available.

Technical Metadata

Technical metadata is used by the Information Systems staff, e.g. systems analysts, data warehouse managers and database administrators. Technical metadata provides a detailed technical blueprint or "wiring diagram" of the data systems that can be used to assist Information Systems expansion and maintenance. Technical metadata traces the flow of data, providing information such as what sources data is extracted from, when the data was extracted, which target it was loaded into and what technical and business rule transformations were applied to that data as it moved from source to target location.

Examples of technical metadata include: what is the system of record for a specific piece of data, what transformations were performed on the source data to produce the target data, what is the structure of the tables and columns in the data warehouse, what is used to reconcile the data with the source system, and when was the last date and time the data was loaded into the target system.

NOTE:

> A mistake many organizations make is assuming that technical metadata can help business end users navigate Information

Systems. Business users need a less technical and more user-friendly way to access and analyze metadata. Business users understand and deal with the data in a different way than technical users.

Why Is Metadata Important

Metadata can provide descriptive information about an information system, e.g. an analyst can use metadata to better understand what information is available and how it is calculated. Metadata provides a detailed analysis of where the information came from and can give a confidence factor for describing the data's validity.

Metadata acts as a road map to the information in your business. Without metadata, business and technical users can access data but not information in context that helps them make business decisions with confidence. Currently, the best practices for new business system development mandate having a metadata strategy that makes the system easy to update and use.

Metadata helps us understand our data and our systems, but more than documentation about how the system runs, it tells us where the system is running and where the physical resources being used by the system are located. With a properly maintained data dictionary, applications become easier to maintain, and, if necessary, replace.

This question is extremely timely given the recent focus on Sarbanes Oxley (a U.S. government information systems reporting act). Answering even basic business systems related questions has become significantly more difficult because of the lack of available, accurate business systems definitions -- metadata.

Why Use a Data Dictionary

Many business systems today use a simple spreadsheet to capture source and target data mappings and conversions. But beyond the project's initial requirements phase, this information quickly becomes outdated and inaccessible to most users who need to interact with data.

Data Dictionary usage for the Data Warehouse

According to Bill Inmon -- "Father of Data Warehousing", metadata is "the description of the structure, content, keys, indexes, etc. of data," (Managing the Data Warehouse, John Wiley & Sons, 1996). More specifically, in a data warehouse environment, metadata can be information about data in the data warehouse, information about how to get a piece of data out of the data warehouse, or information about the quality of data in the data warehouse. Metadata can even give information about how to run warehouse tools to perform different tasks. Metadata about a data warehouse includes information about systems, processes, source and target databases, data transformations, data cleansing, data access, datamarts, and Business Intelligence tools.

For example, a user looking at a regional sales report might think that Total Sales includes all discounts and shipping and handling charges. But Total Sales might not include any or all of these elements. To help the end user, the data warehouse needs metadata that ties the reports' columns to data transformations, data queries, field calculations, and source database tables and columns.

Data Dictionary usage for Application Integration

Leveraging the power of the data dictionary within your systems and applications will ease your company's foray into application integration. An emerging area that is heavily reliant upon the availability of metadata is application integration. Moving forward, metadata can significantly increase your ability to deliver personalized data to customers and business partners. In the age of e-commerce, clearly one of the defining factors is the ability to customize delivery of a singular set of information to multiple recipients in a variety of formats. For example, for a large bank to integrate their investment systems with their retail banking systems, it is necessary to understand the data, data types, and data sources for both systems. Through integration of these systems, customers can be provided with a consolidated statement, instead of two separate statements from the same bank. Or, in the case of a Web interface, the consolidated statement can simplify navigation by not requiring the user to view their checking and investment account information separately. In both of these cases, it is the underlying metadata that will drive the integration that facilitates the personalized delivery of information, and thus, provides a more professional impression of the bank to the customer.

Enterprise-level decisions are especially dependent upon consistent definitions. The recent proliferation of independent datamarts (departmental reporting databases) has underscored the importance of ensuring data consistency. Organizations that have not considered key data (e.g. customer account identifiers) while implementing data systems are likely to find themselves with large amounts of disparate data that can't be shared or combined. Some organizations that have built these "stovepipe" decision support systems are finding that they are incapable of making accurate decisions across the enterprise because they have no way to consistently define the data. By documenting key data while implementing independent datamarts can prevent this problem.

Data Dictionary usage for Process Reuse

Additionally, the same metadata that drives personalization of information can drive data reuse. When a company has a thorough understanding of the data it has, it can then intelligently decide the data's overall benefit to the company. More importantly, when metadata is made available to all corporate personnel, new and innovative ways to use the data can emerge. For example, if the Information Systems department is the only group that has access to the metadata, innovation can be constrained. However, when the business manager for new account development has access to a source of well-defined metadata, then that person is empowered to devise new campaigns for attracting new customers.

Of note, reuse also leads to lower costs for software development, implementation, and maintenance, and increases the opportunity for standardization of information across the company. The latter point is extremely important for companies looking to optimize their internal processes or to create straight-thru processing.

Making Metadata Accessible

Part of a company's commitment to capturing metadata requires two additional decisions. The first is where the metadata will be stored, and the second is how the metadata will be made available to those who need it.

In terms of storage and access, the most obvious answer is to use a metadata repository -- a data dictionary like SuperLuminate. This is a specialized database application designed to provide the infrastructure and support for storage of interrelated components of information. As stated earlier, very few, if any, metadata components stand on their own. Metadata repositories not only help capture information about singular metadata components, but also about the relationships between individual components. Metadata repositories also provide important functionality for searching and browsing the available metadata, delivering one of the more important functions -- producing impact analyses.

Impact analyses identifies all the resources that rely on a particular system component and, therefore, assists in defining all the resources that would be impacted by a change in a system component. Producing these types of reports however, requires inputting and maintaining the necessary information in a data dictionary.

Data Dictionary Definition References

These articles may seem dated but their content is still very relevant.

Shiraz Kassam	Freedom of Information	April 2002

Patrick Cross, Saeed Rahimi	Using the Microsoft Repository	March 2000
JP Morgenthal, Priscilla Walms	Mining for Metadata	Feb 2000
David Gleason V.P. Platinum	Business Metadata	May1999

User Quick Start (online demo)

SuperLuminate (from SuperLuminate help menu)

This chapter refers to the online demo at www.superluminate.com

This basic help window should provide a new user with a few examples to help them navigate through the SuperLuminate data dictionary application for the first time. For more advanced help please refer to the user guide or connect to the SuperLuminate web site at www.superluminete.com.

Notation Convention

You can determine the SuperLuminate page you are viewing by checking the message bar on that page. If the message bar reads "[Login] Please Login", you are on the [Login] page.

In the following conventions will be used:

- Square brackets [ ] will indicate a page, e.g. [Login] is the login page
- Wavy brackets { } indicate something that can be pressed (selected) including, buttons, dropdowns, and hyperlinks
- Round brackets ( ) are used only to restate or enlighten an example

Basic Look and Feel

- SuperLuminate [Login] page => Press the {Read Only Access} button. This will log you into SuperLuminate as a read only user. You will be able to select and see all "public" records and relationships, but you will not be able to change any records or system settings.

- On the [Main Menu] page à Press the {Search} button. This will drop you into the [Search] page where you can select records. The [Search] page layout consists of all of the fields that makeup a basic record in the SuperLuminate data dictionary. This does not include extended properties or relationships.

- Anytime you get lost à Press the {Main Menu} hyperlink found on the top right hand side of all pages. From the [Main menu] page you can start again or logoff.

Selection by Taxonomy

On the top of the [Search] page you will see a set of six properties:

- Category -- Highest level of segmentation, e.g. Department, Project
- Subject -- Defines the schema or schemas
- Class -- Defines the Objects (entities) in a schema
- Type -- Can be used to define state or status, e.g. TST, PRD
- Name -- The name of the object (unique within Class)
- Version -- Allows Objects to be versioned saving history

These six properties form the identifier (key) for all records contained in SuperLuminate.

The properties Category, Subject, Class, and Type, define the taxonomy (ordered relationship) of the record. Any of the four taxonomy dropdown properties (Category, Subject, Class, and Type) can be selected and searched.

- First select the {Subject} property dropdown list button and pick the {Business} list entry.
- Now press the {Search Now} button. This will display a list of all records in the SuperLuminate data dictionary that have been assigned the Subject value "Business".
- To select the contents of a specific record, press a button under the "SELECT" label that corresponds to the record you want to view.
- On the [Selection] page, pressing the {Search Again} button will return you to the [Search] page.

Selection by Entry

On the [Search] page below the taxonomy properties, Category, Subject, Class, and Type, you will see the fields labeled, Name, Alias, and Description. Note that the label names are followed by a "red diamond" (•). This indicates that when the corresponding field is searched for the word or phrase you entered, the word or phase will be found no matter where in the field it is located and is not dependent on the case of the characters, e.g. uppercase and lowercase will be treated as the same. If you search for “Table” you will find “Table” and “TABLE”.

- Type "customer" into the field labeled Description.
- Press the {Search Now} button. This will display all records that you have authorization to access in the SuperLuminate data dictionary that have the word "customer" in the Description field.
- To select a specific record, press a button under the "SELECT" label to select and display the corresponding record.
- Pressing the {Search Again} button will return you to the [Search] page.

Compound Selection

On the [Search] page, for this example, we will select a taxonomy value and enter a search word. Adding more entries to the selection criteria will reduce the set of records returned. You can specify as many criteria as you wish.

Note: If no records meet the criteria you have set, no records will be returned.

- Select the {Class} property dropdown list button and pick the {Table} list entry.
- Type "user" into the field labeled Description.
- Press the {Search Now} button. This will display a list of all records in the SuperLuminate data dictionary with a Class value of "Table" that have the word "user" somewhere in the Description field. Remember only the records that you are authorized to see will be returned.
- To select a specific record, press a button under the "SELECT" label to select and display the corresponding record.
- Pressing the {Search Again} button will return you to the [Search] page.

Following the Parent Child Relationship Links

For this example, we will first retrieve the records selected in the last example then we will select a specific record and then see how that record is linked hierarchically to other records of different types. We will be able to travel from record to record up and down the tree of relationships that was defined in SuperLuminate by the administrator. The relationship tree, technically known as a schema (A.K.A. Information Model) is a key feature of SuperLuminate and is discussed in far more detail in the user guide.

- Select the {Class} property dropdown list button and pick the {Table} list entry.
- Type "user" into the field labeled Description.
- Press the {Search Now} button.
- Press the button under the "SELECT" label corresponding to the record with the Name "SL_Prod.SL_Object".

On the [Selection] page you should now be viewing the record named "SL_Prod.SL_Object". Using the elevator viewing bar on the side of your browser you can view the contents of the entire record. You can see the record is broken into sections:

- Base Properties -- Key properties and descriptions
- Control Properties -- Every record contains ownership, dates, status
- Extended Properties -- Properties added by your administrator
- Parent Objects -- Objects that a hierarchically above this Object
- Child Objects -- Objects that a hierarchically below this Object

Extended Properties are properties that have been defined by the administrator as extensions to SuperLuminate.

Parent Objects are records that fall higher on the hierarchical tree than the current record and Child Objects are records that fall lower on the tree.

Schema (A.K.A. Information Model): In this example we are currently looking at a record that has a Class value of "Table".

The example schema enforces relationships between Classes as follows:

- An Application Object can be a parent to many Database Objects
- A Database Object can be a parent to many Table Objects
- A Table Object can be a parent to many Column Objects

Because we are currently viewing a "Table" record. The Parent Object will be of Class "Database" and the Child Objects will be of Class "Column". You can traverse this tree by selecting a Parent Object or a Child Object to view the corresponding record.

You can repeat this action again and again moving from record to record through the relationship tree (schema).

- Select parent name "SL_Prod" to see the database record.
- Select child name "SL_Prod.SL_Object" to return to the previous table record.
- Select child name "SL_Prod.SL_Object.Desc_TX" to view one of the column record. Note: In the column record is a set of Extended Properties that has been defined for this Class of record by the administrator.

Navigation Architecture

Notation

The following notation will be used in this manual to help you with navigation:

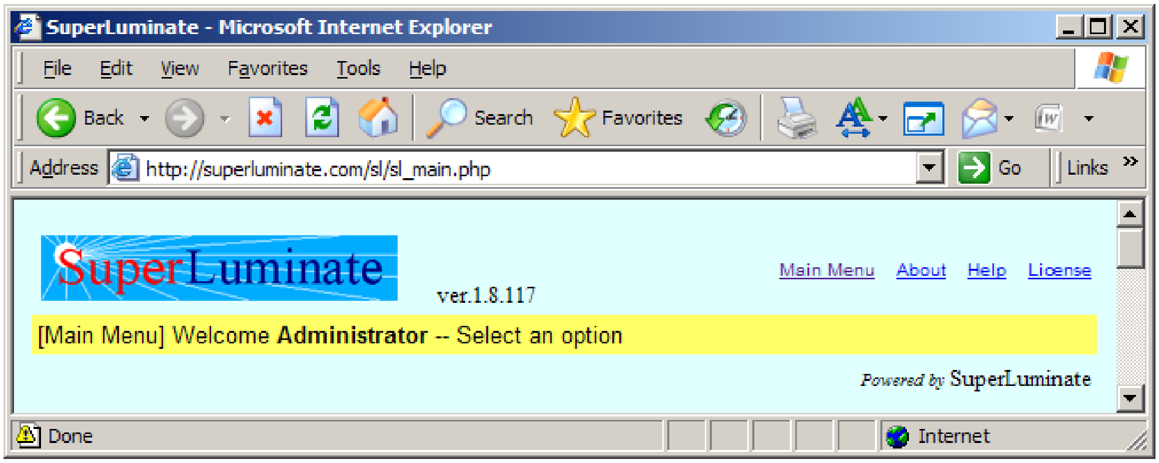
- All pages are identified in the text by square brackets [ ], e.g. [Login]
- All buttons are identified in the text by wavy brackets { }, e.g. {User login}
- All drop down picklists are identified in the text by wavy brackets { }, e.g. the {Category} picklist on the [Select] page
- All hyperlinks – text that you can click on to produce an action – are identified by wavy brackets { } usually followed by the word hyperlink, e.g. {Help} hyperlink on the [Main Menu] page
- Navigation notation will be seen at the top of some chapters to indicate when in the SuperLuminate application we are, e.g.

  [Login] à [Main Menu] à [Search]

  The above indicates you are on the [Search] page that was arrived at by first being on the [Login] page then selecting the [Main Menu] page and then selecting the [Search] page.

Page Header Menu

All pages have a header menu.


SuperLuminate - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://superluminate.com/sl/sl_main.php Go Links
SuperLuminate
ver.1.8.117
Main Menu About Help License
Powered by SuperLuminate
Done
Internet

[Main Menu] Welcome **Administrator** -- Select an option

The header menu, on all SuperLuminate pages, contains the following items:

- {Main Menu} hyperlink
  - Hyperlink to return to [Main Menu] page where you can select another function or logoff

- {About} hyperlink
  - Hyperlink to display a popup window that contains a description of the SuperLuminate data dictionary application

- {Help} hyperlink
  - Hyperlink to display a popup window that contains basic help to get new users started

- {License} hyperlink
  - Hyperlink to display a popup window that contains license Information: SuperLuminate is free for your use and you may also distribute SuperLuminate for free. You cannot sell SuperLuminate or any part of SuperLuminate

- Message Bar
  - The message bar at the top of each page displays the page name in the square brackets, e.g. [Login] for the login page. After the page name is an informational message that asks for input or displays the result of a previous function

- Version
  - Displays the current version of the SuperLuminate application that you are running. The parts of the version number are (major).(minor).(release)

- Logo

  - By default the logo is the SuperLuminate logo. This logo can be changed to your company logo by your administrator via the configuration page.

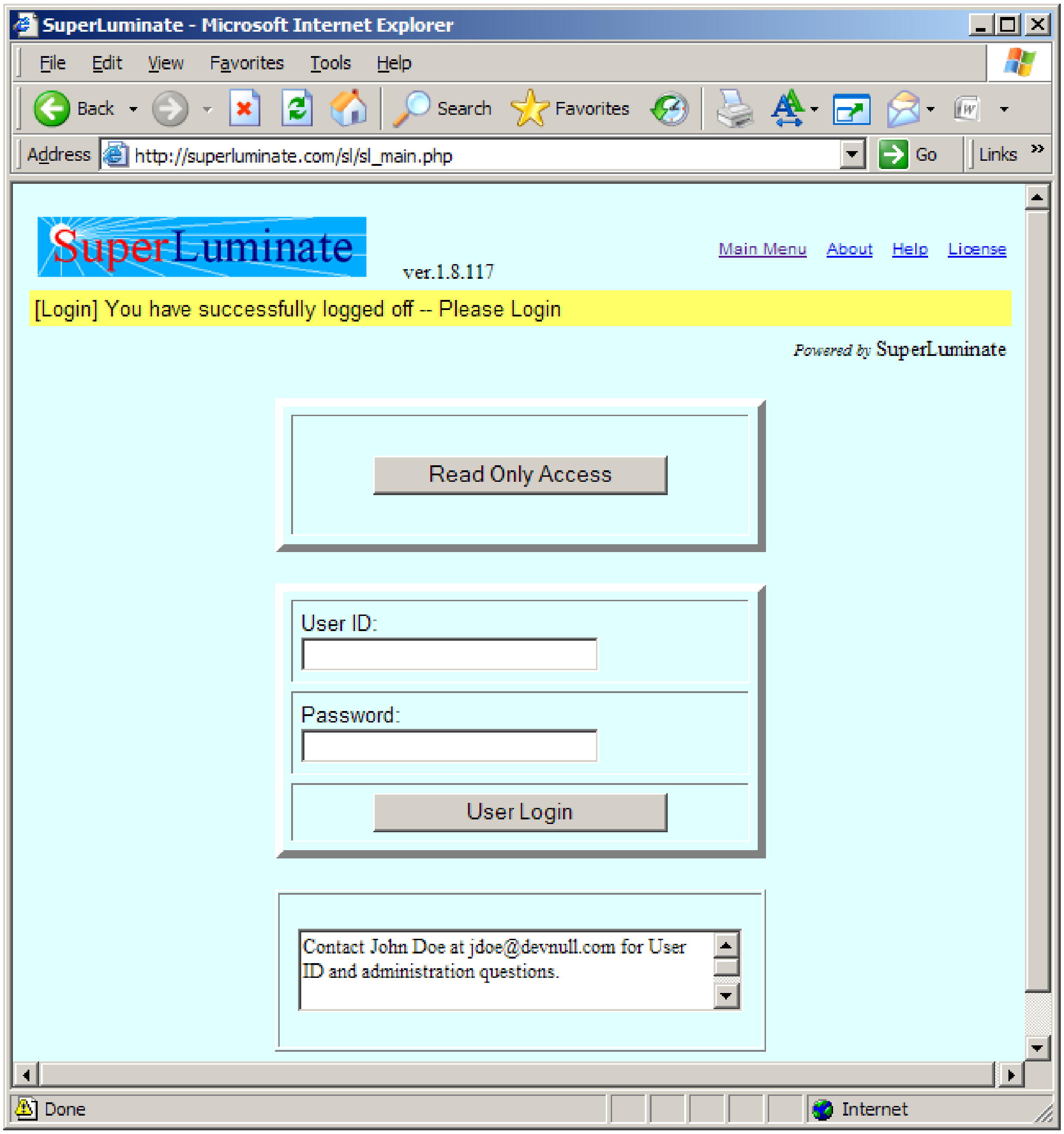
- Powered by SuperLuminate
  - Developer acknowledgment

Logon Page

[Login]

Main Login Page: The read only access button can be configured "Off" or "On" by your SuperLuminate administrator. Read only access does not require a User ID or Password but has limited functional access.

Select {Read Only User} to browse records that have been authorized for access by a read only user


SuperLuminate - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://superluminate.com/sl/sl_main.php Go Links
SuperLuminate
ver.1.8.117
Main Menu About Help License
Powered by SuperLuminate
Read Only Access
User ID:
Password:
User Login
Done
Internet

[Login] You have successfully logged off -- Please Login

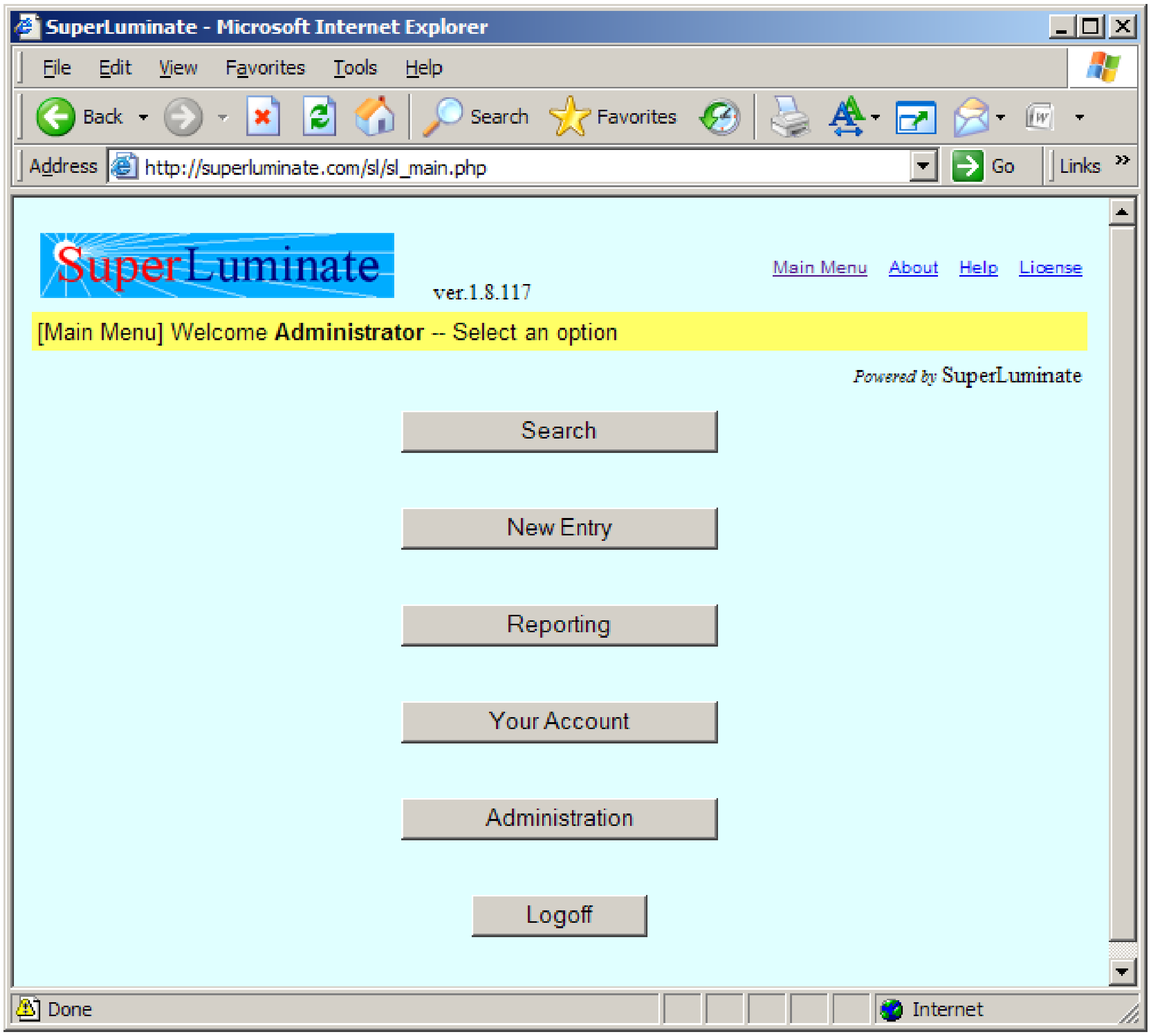
Contact John Doe at jdoe@devnull.com for User ID and administration questions.

Main Menu

[Login] à [Main Menu]

This page is the main menu. The main menu is the pivotal page in the application – all pages branch from this page.

Select {Search} to search the SuperLuminate database. Note: We are logged on as the “Administrator” – full access to SuperLuminate functionality.



Address: http://superluminate.com/sl/sl_main.php

[Main Menu] Welcome **Administrator** -- Select an option

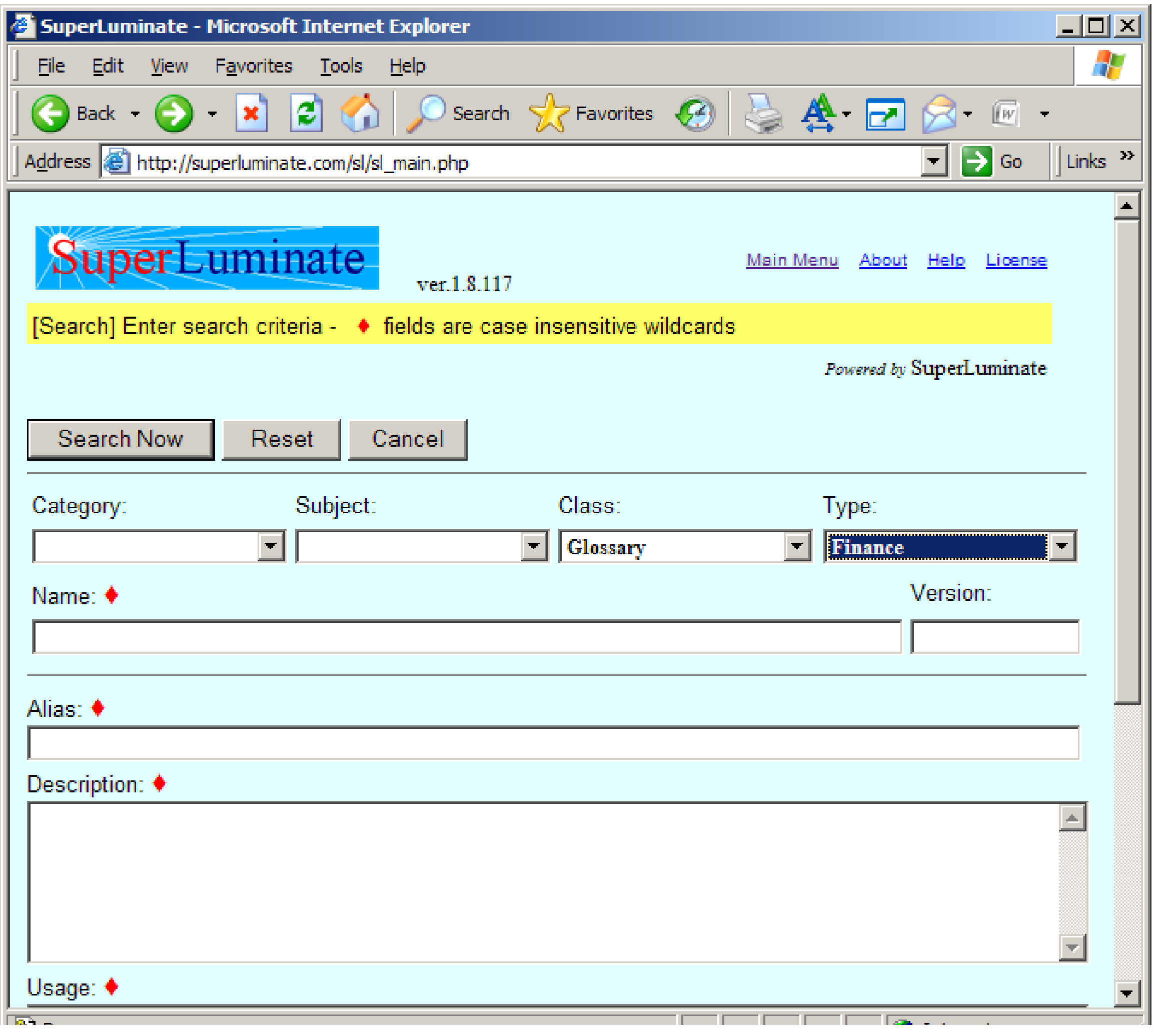
Search Page (search example #1)

[Login] à [Main Menu] à [Search]

Note: The example metadata load files that accompanied the SuperLuminate installation zip file must be loaded by your SuperLuminate administrator to perform the following search examples. If they have not loaded the metadata to your implementation you can still perform these tests using the SuperLuminate demo at www.SuperLuminate.com.

Press the {Class} drop down picklist button and select the "Glossary" list entry. Then press the {Type} drop down picklist button and select the "Finance" list item. Then press the {Search Now} button to retrieve all records in the SuperLuminate database that match the selected criteria.

Another example: leave the taxonomy picklists blank and type the word “customer” in the Name field. Then press the {Search Now} button. The fields denoted by the diamonds are not case sensitive.



http://superluminate.com/sl/sl_main.php

ver.1.8.117

[Search] Enter search criteria - ♦ fields are case insensitive wildcards

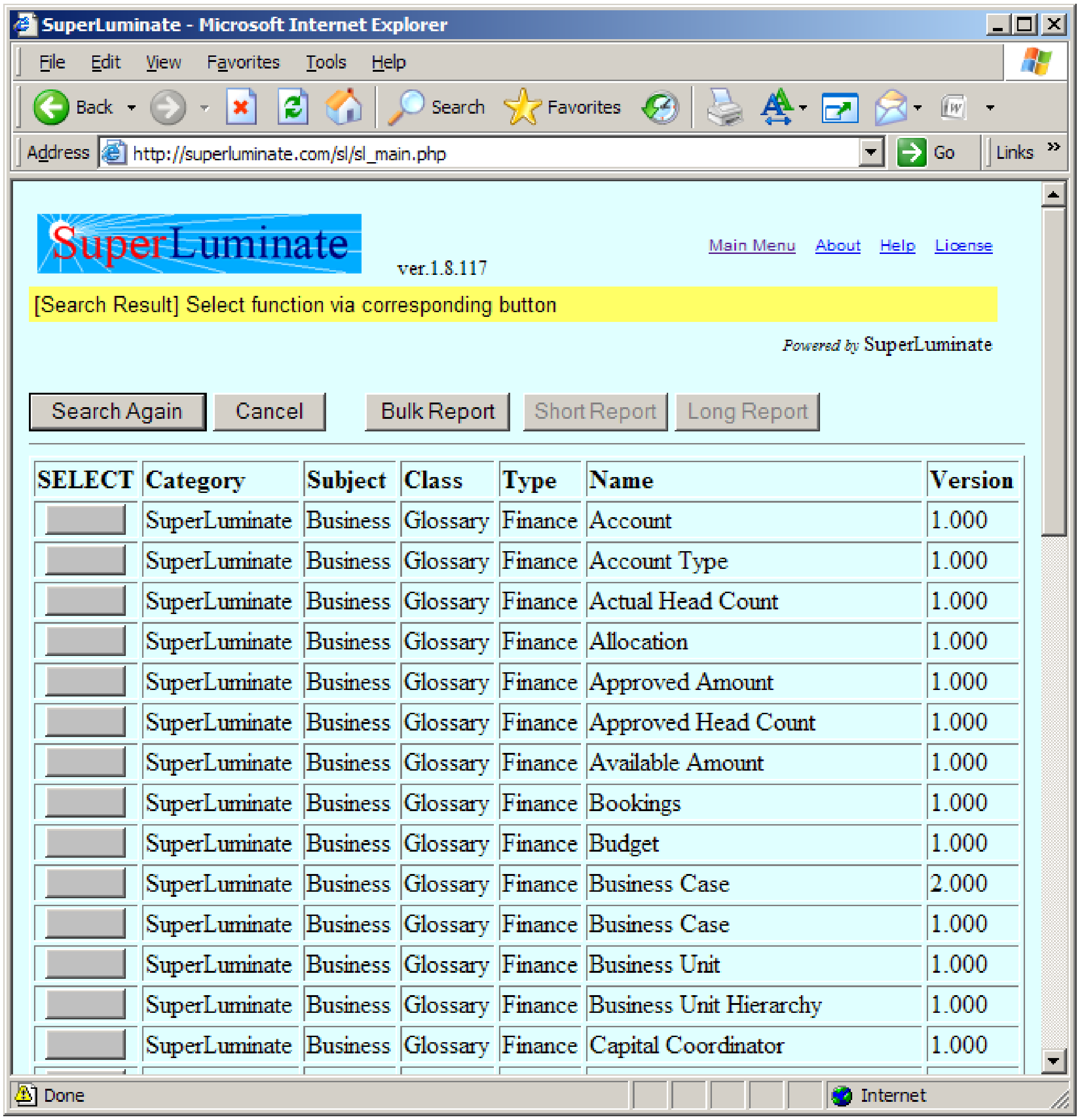
Class: Glossary

Type: Finance

Search Results Page (search example #1)

[Login] à [Main Menu] à [Search] à [Search Result]

This page displays a list of all the records that matched the search criteria you specified. Under the header name "SELECT" press the button corresponding to the record with the Name "Business Unit Hierarchy". You may need to scroll down the list.


SuperLuminate - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://superluminate.com/sl/sl_main.php Go Links
SuperLuminate
ver.1.8.117
Main Menu About Help License
Powered by SuperLuminate
Search Again Cancel Bulk Report Short Report Long Report
Done
Internet

[Search Result] Select function via corresponding button

SELECT	Category	Subject	Class	Type	Name	Version
☐	SuperLuminate	Business	Glossary	Finance	Account	1.000
☐	SuperLuminate	Business	Glossary	Finance	Account Type	1.000
☐	SuperLuminate	Business	Glossary	Finance	Actual Head Count	1.000
☐	SuperLuminate	Business	Glossary	Finance	Allocation	1.000
☐	SuperLuminate	Business	Glossary	Finance	Approved Amount	1.000
☐	SuperLuminate	Business	Glossary	Finance	Approved Head Count	1.000
☐	SuperLuminate	Business	Glossary	Finance	Available Amount	1.000
☐	SuperLuminate	Business	Glossary	Finance	Bookings	1.000
☐	SuperLuminate	Business	Glossary	Finance	Budget	1.000
☐	SuperLuminate	Business	Glossary	Finance	Business Case	2.000
☐	SuperLuminate	Business	Glossary	Finance	Business Case	1.000
☐	SuperLuminate	Business	Glossary	Finance	Business Unit	1.000
☐	SuperLuminate	Business	Glossary	Finance	Business Unit Hierarchy	1.000
☐	SuperLuminate	Business	Glossary	Finance	Capital Coordinator	1.000

Selection Display Page (search example #1)

[Login] à [Main Menu] à [Search] à [Search Result] à [Selection]

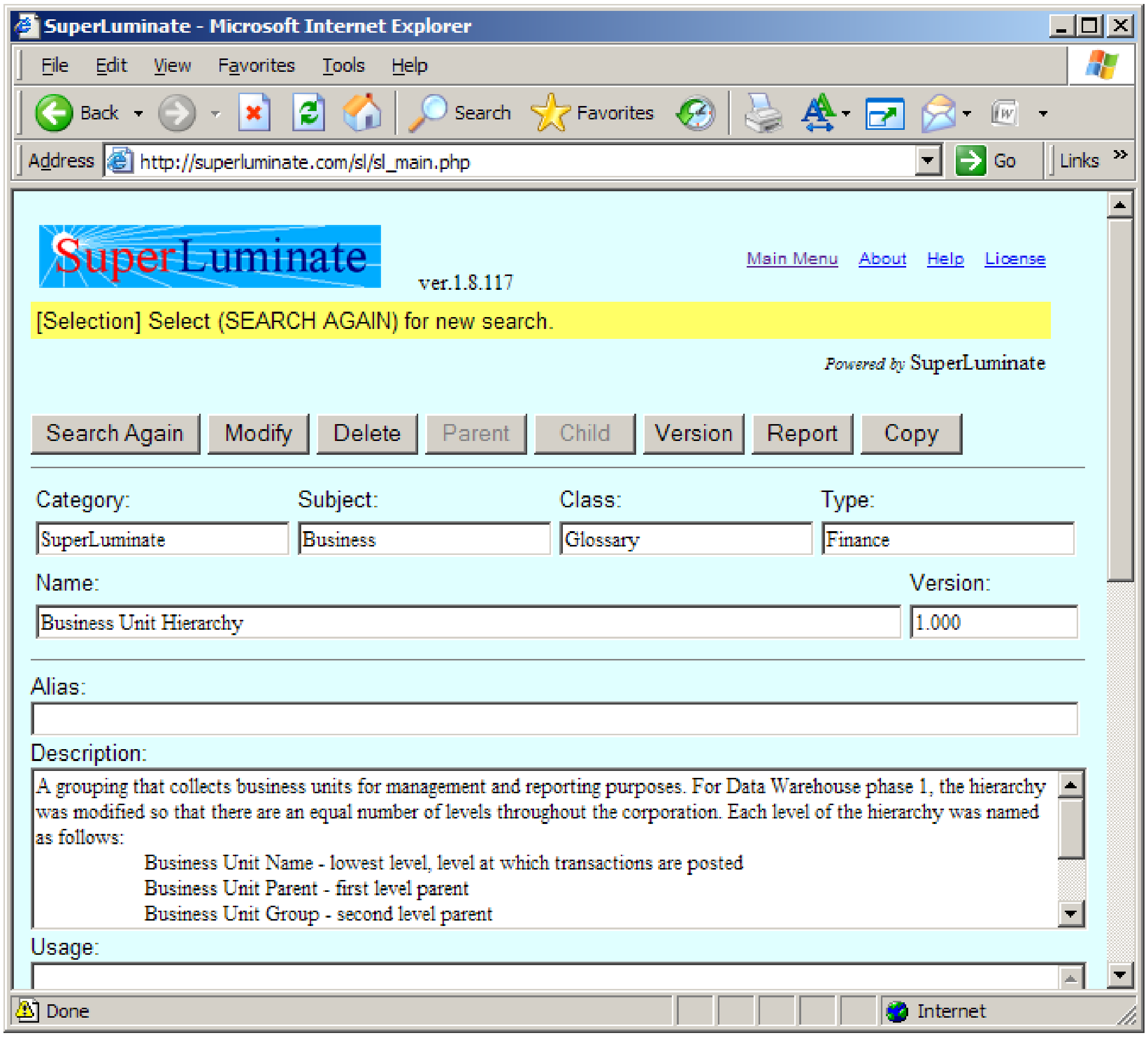
This page displays the information that is being maintained for the object "Business Unit Hierarchy".

Note:

Scroll down and see at the bottom of the page you can see that the object contains no External Properties or Parent/Child Relationships to other objects.

You can press the {Report} button to see a print ready formatted report; then press the browser back button to return to this page.

The buttons that are "low-lite" are to functions that are not available to you for this record (object).



http://superluminate.com/sl/sl_main.php

ver.1.8.117

[Selection] Select (SEARCH AGAIN) for new search.

Category: SuperLuminate
Subject: Business
Class: Glossary
Type: Finance
Name: Business Unit Hierarchy
Version: 1.000
Alias:
Description:

A grouping that collects business units for management and reporting purposes. For Data Warehouse phase 1, the hierarchy was modified so that there are an equal number of levels throughout the corporation. Each level of the hierarchy was named as follows:

Business Unit Name - lowest level, level at which transactions are posted
Business Unit Parent - first level parent
Business Unit Group - second level parent

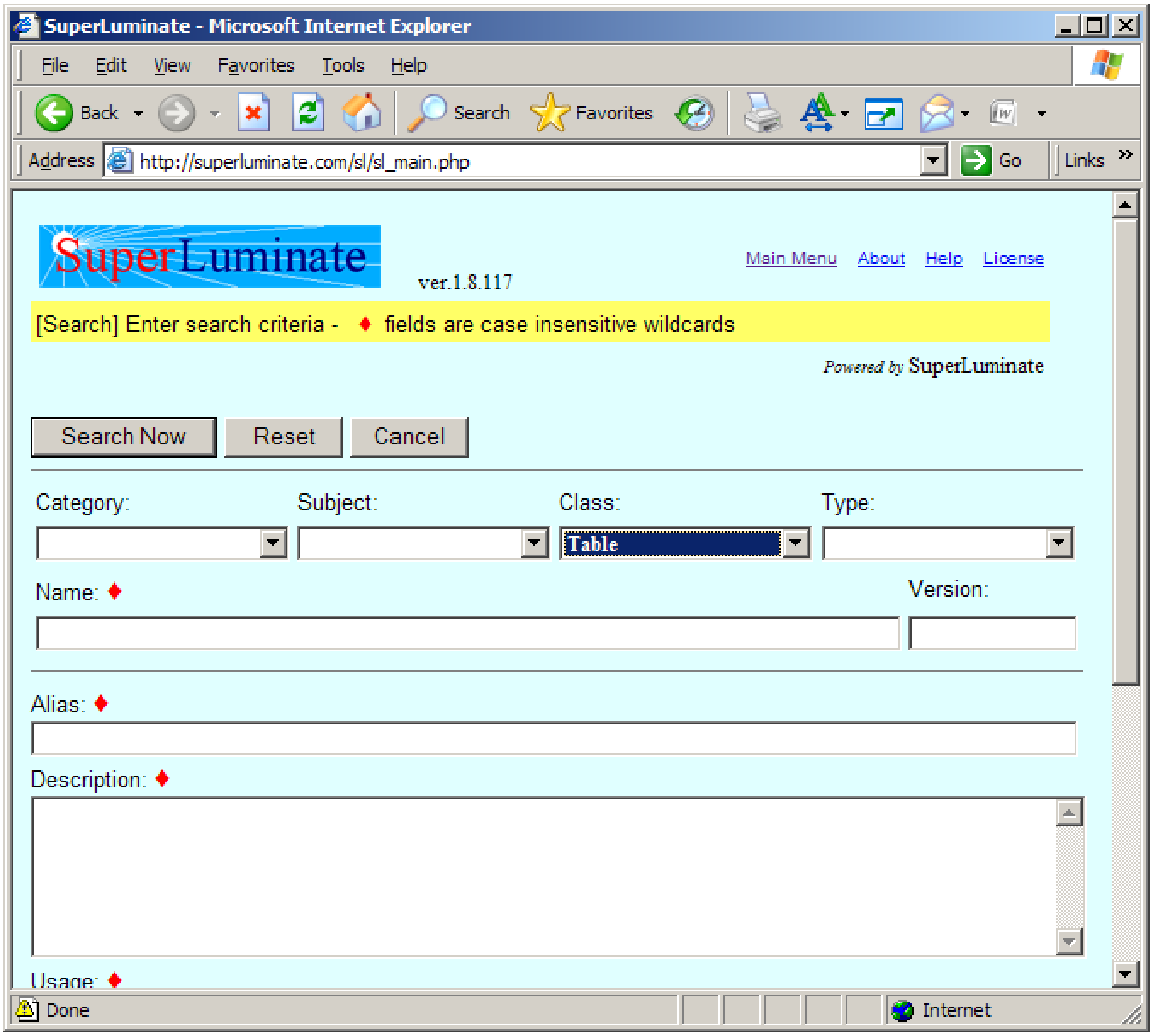
Usage:

Search Page (search example #2)

[Login] à [Main Menu] à [Search]

Starting a new search – from the previous search page to start a new search press the {Search Again} button or press the {Main Menu} hyperlink and then press the {Search} button on the [Main Menu].

Press the {Class} drop down picklist button and select the "Table" list entry. Then press the {Search Now} button to retrieve a list of all records in the SuperLuminate database that match the selected criteria.



Address: http://superluminate.com/sl/sl_main.php

[Search] Enter search criteria - ♦ fields are case insensitive wildcards

Category:
Subject:
Class: Table
Type:
Name: ♦
Version:
Alias: ♦
Description: ♦
Usage: ♦

Search Results Page (search example #2)

[Login] à [Main Menu] à [Search] à [Search Result]

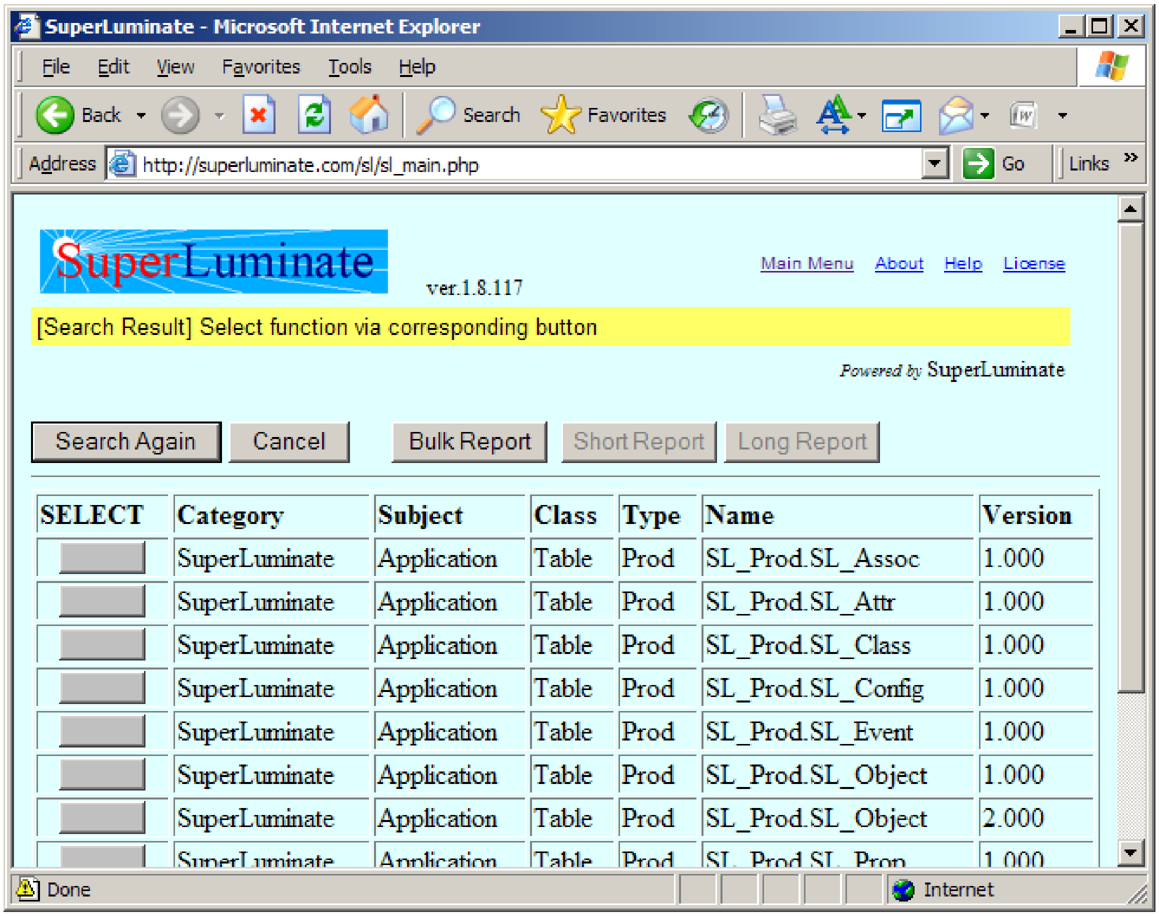
Select the record with the Name: "SL_Prod.SL_Object".

Note:

The {Bulk Report} button creates a formatted report that can be cut and pasted into a spreadsheet containing all fields for the selected objects.

If the search criteria you specified do not match any records, no records will be displayed on this page. If no records show up, press the {Search Again} button and try to re-specify the search criteria.

A data dictionary can contain 100s of thousands of records in a small implementation. If you specify the criteria: picklist {Class} "Column" thousands of rows might match the criteria. The SuperLuminate default is to only allow 2000 records to be retrieved in a search (this can be adjusted by your SuperLuminate administrator). If your search retrieves more than 2000 records you will be returned to the [Search] page and asked to further specify your criteria thus reducing the number of matching records.



[Search Result] Select function via corresponding button

SELECT	Category	Subject	Class	Type	Name	Version
☐	SuperLuminate	Application	Table	Prod	SL_Prod.SL_Assoc	1.000
☐	SuperLuminate	Application	Table	Prod	SL_Prod.SL_Attr	1.000
☐	SuperLuminate	Application	Table	Prod	SL_Prod.SL_Class	1.000
☐	SuperLuminate	Application	Table	Prod	SL_Prod.SL_Config	1.000
☐	SuperLuminate	Application	Table	Prod	SL_Prod.SL_Event	1.000
☐	SuperLuminate	Application	Table	Prod	SL_Prod.SL_Object	1.000
☐	SuperLuminate	Application	Table	Prod	SL_Prod.SL_Object	2.000
☐	SuperLuminate	Application	Table	Prod	SL_Prod.SL_Prop	1.000

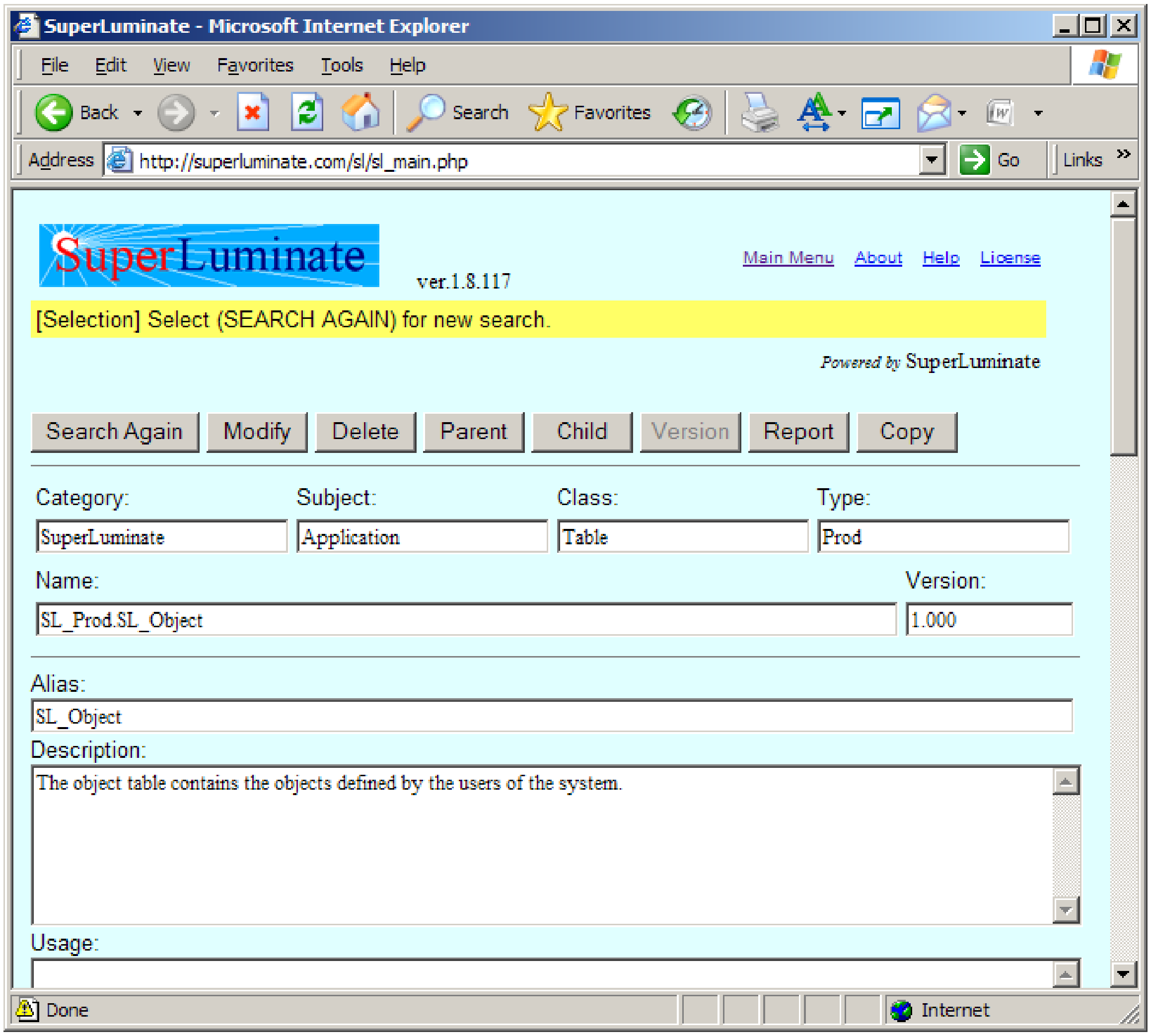
Selection Display Page (search example #2)

[Login] à [Main Menu] à [Search] à [Search Result] à [Selection]

Top half of "SL_Prod.SL_Object" record that contains the descriptive information for the retrieved object.

Note:

If you press the {Report} button, a printer friendly "formatted" record will be displayed containing all of the information expanded. Press the browser back button to return to this page.



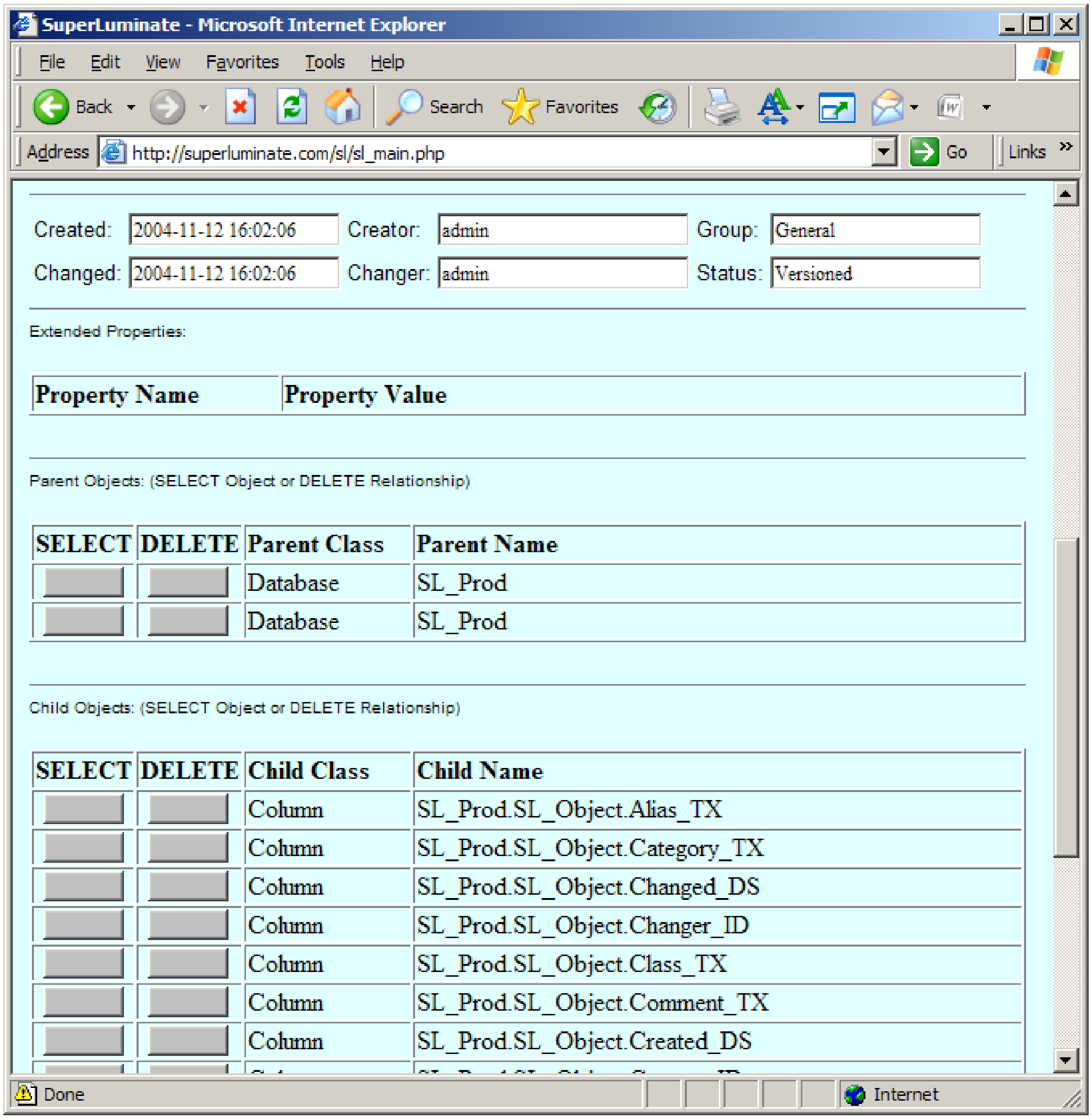
[Selection] Select (SEARCH AGAIN) for new search.

Category: SuperLuminate
Subject: Application
Class: Table
Type: Prod
Name: SL_Prod.SL_Object
Version: 1.000
Alias: SL_Object
Description: The object table contains the objects defined by the users of the system.
Usage:

Bottom half of "SL_Prod.SL_Object" record containing Control Attributes, Extended Properties and Relationships to other Objects.

Note:

> On this page you will see the {Delete} button activated. The administrator and or the owner of the record can delete relationships. These buttons will be “low-lite” for all other users.


SuperLuminate - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Address
http://superluminate.com/sl/sl_main.php
Go
Links
Done
Internet

Created: 2004-11-12 16:02:06 Creator: admin Group: General

Changed: 2004-11-12 16:02:06 Changer: admin Status: Versioned

Extended Properties:

Property Name	Property Value

Parent Objects: (SELECT Object or DELETE Relationship)

SELECT	DELETE	Parent Class	Parent Name
		Database	SL_Prod
		Database	SL_Prod

Child Objects: (SELECT Object or DELETE Relationship)

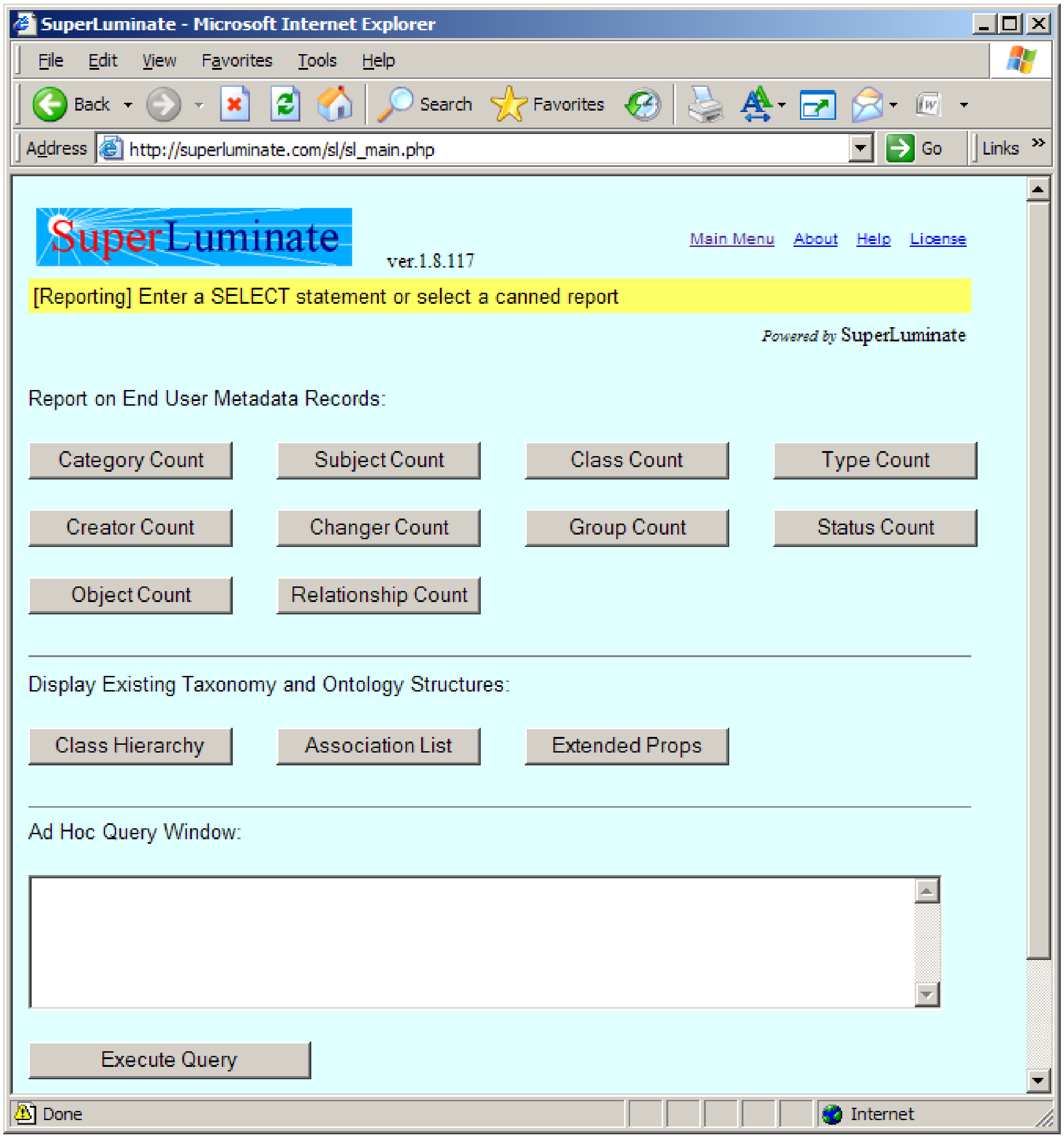
SELECT	DELETE	Child Class	Child Name
		Column	SL_Prod.SL_Object.Alias_TX
		Column	SL_Prod.SL_Object.Category_TX
		Column	SL_Prod.SL_Object.Changed_DS
		Column	SL_Prod.SL_Object.Changer_ID
		Column	SL_Prod.SL_Object.Class_TX
		Column	SL_Prod.SL_Object.Comment_TX
		Column	SL_Prod.SL_Object.Created_DS

Generalized Reporting

[Login] à [Main Menu] à [Reporting]

This page selected from the [Main Menu] will allow a user to select a sample set of canned reports via the push of a button. Your administrator can add new canned reports as required. If the administrator allows a user to create and run custom SQL SELECT queries this can be done via the Ad Hoc SQL Query window on this page.

NOTE: The administrator can run any form of SQL statement via the Ad Hoc SQL Query window but a user can only run a SELECT query.



[Reporting] Enter a SELECT statement or select a canned report

Report on End User Metadata Records:

Display Existing Taxonomy and Ontology Structures:

Ad Hoc Query Window:

SuperLuminate Lite (Glossary Read Only Access)

[Search Lite]

If SuperLuminate Lite is made available to the end user by the administrator (via the configuration page) you will be able to access the glossary directly by:

http:// (your website address) /sl/sl_main.php?Glossary

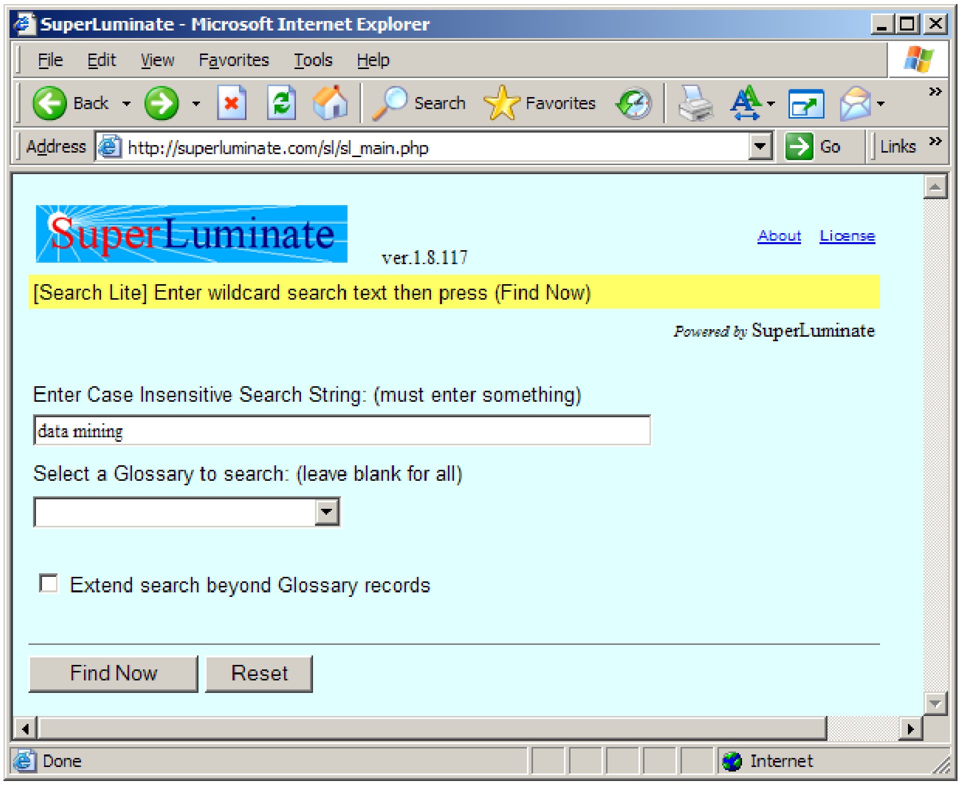
Administrator Note:

You can use the following URL format can be embedded in a button to control the size and placement of the SuperLuminate Lite window. This URL format will also force the window to the front if it is covered by another window.

<a href="sl/sl_main.php?Glossary" onclick="var newWin = window.open(href, 'WinName', 'height=500, width=650, left=75, top=25, scrollbars=yes'); newWin.focus(); return false;">Data Dictionary (Lite) -- Read Only Glossary Access</a>

SuperLuminate Lite provides out-of-the-box easy end user access to glossary (Class = “Glossary”) terms.

Note: By checking the {Extend Search} checkbox the request will be searched for across all objects classes not just object classes equal to “Glossary.”


SuperLuminate - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://superluminate.com/sl/sl_main.php Go Links
SuperLuminate
ver.1.8.117
About License
Powered by SuperLuminate
Find Now Reset
Done
Internet

[Search Lite] Enter wildcard search text then press (Find Now)

Enter Case Insensitive Search String: (must enter something)

data mining

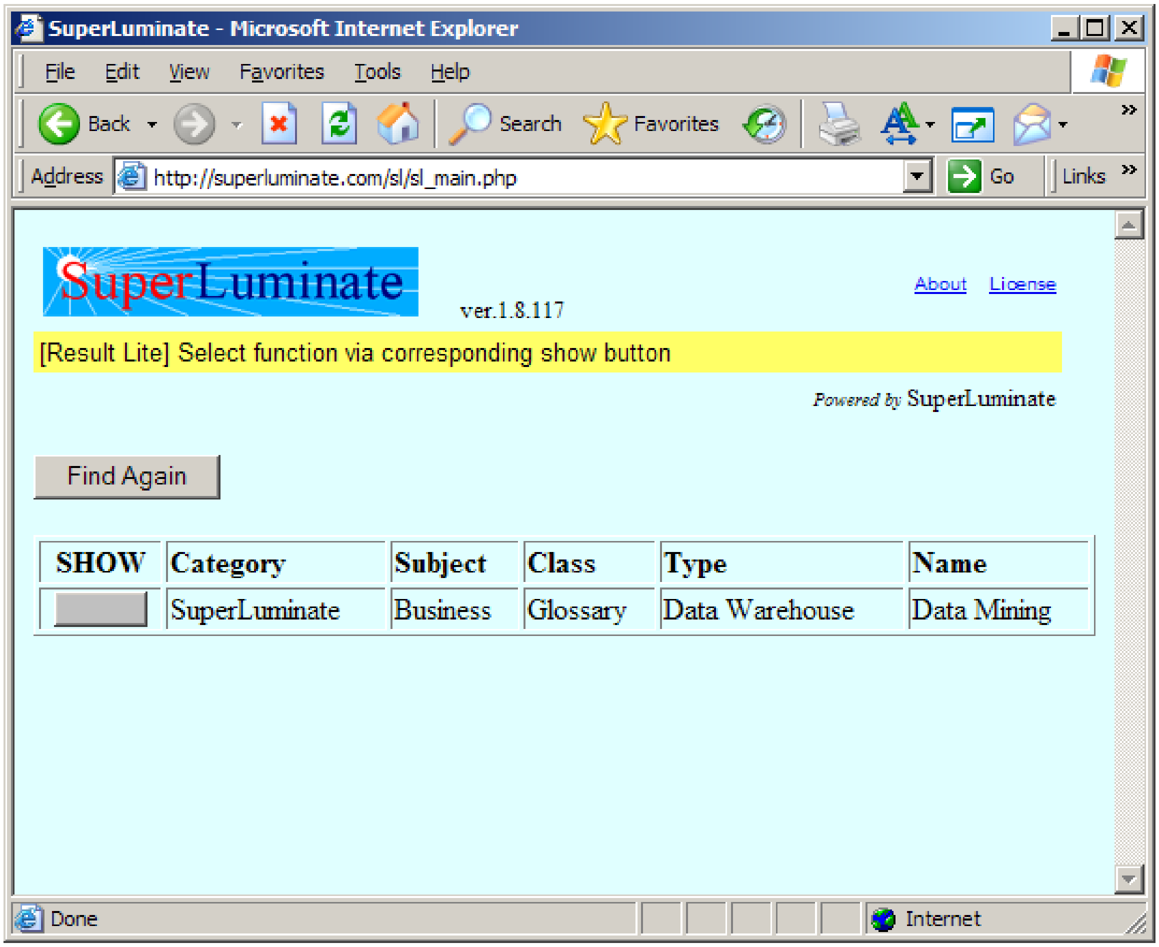
Select a Glossary to search: (leave blank for all)

☐ Extend search beyond Glossary records

SuperLuminate Lite (glossary) selection page:

[Search Lite] à [Result Lite]

Under the “SHOW” header press the button corresponding to record with the Name “Data Mining”.



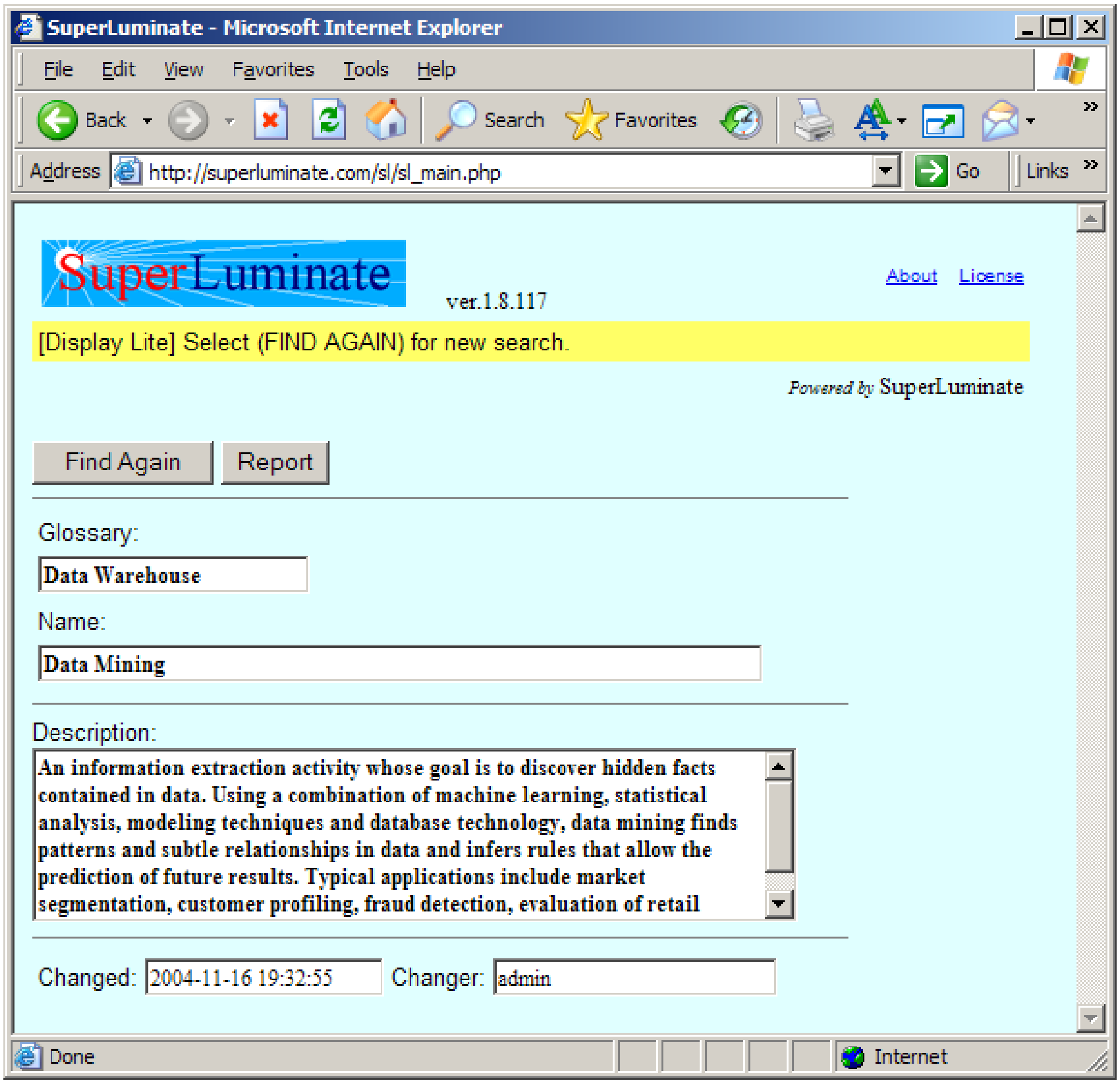
[Result Lite] Select function via corresponding show button

SHOW	Category	Subject	Class	Type	Name
☐	SuperLuminate	Business	Glossary	Data Warehouse	Data Mining

SuperLuminate Lite (glossary) display page:

[Search Lite] à [Result Lite] à [Display Lite]

NOTE: You are able to create a report for any glossary term selected. The report will contain all of the information that is contained in the object record along with extended attributes.



http://superluminate.com/sl/sl_main.php

ver.1.8.117

[Display Lite] Select (FIND AGAIN) for new search.

Glossary: Data Warehouse

Name: Data Mining

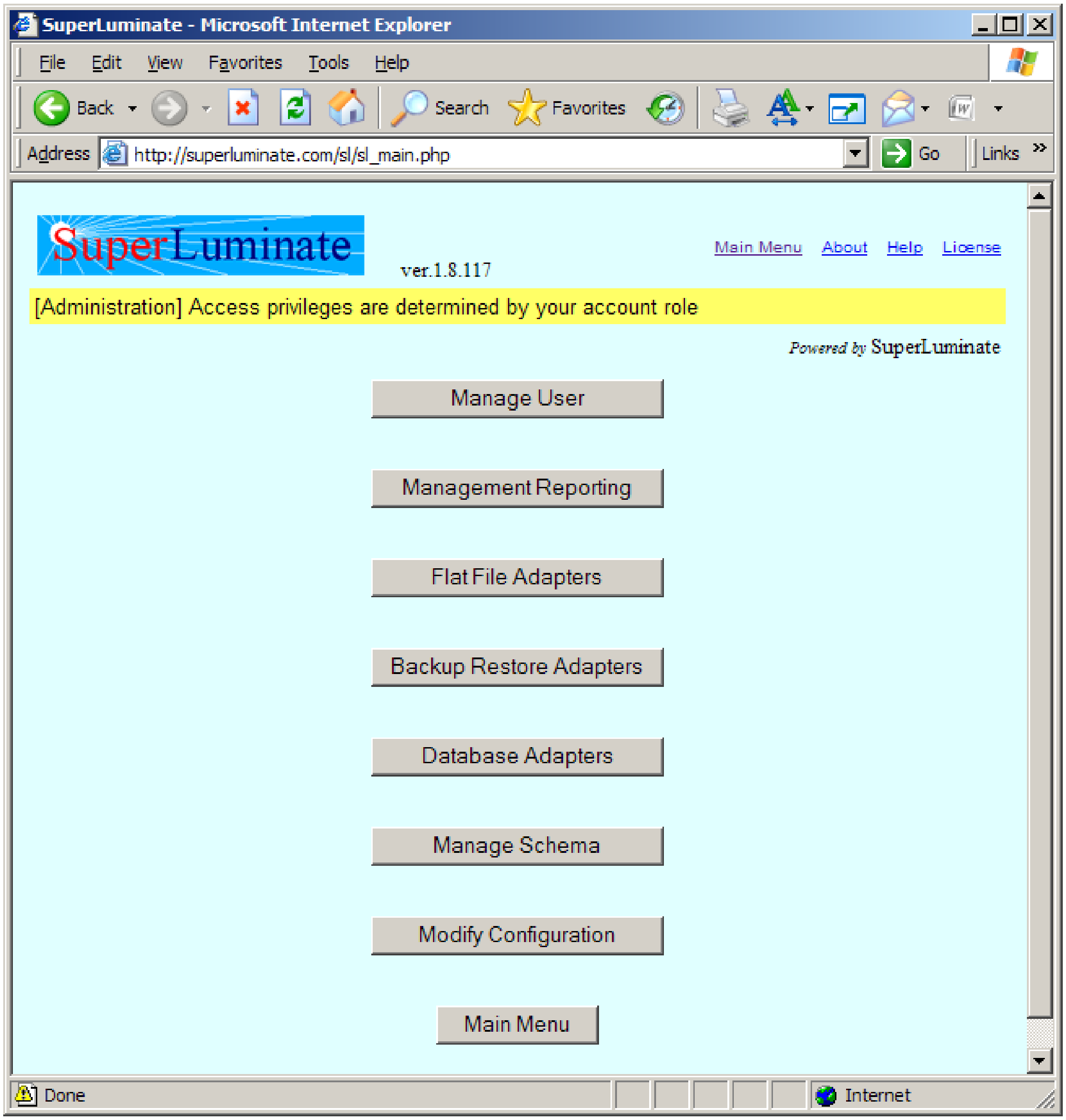
Description: An information extraction activity whose goal is to discover hidden facts contained in data. Using a combination of machine learning, statistical analysis, modeling techniques and database technology, data mining finds patterns and subtle relationships in data and infers rules that allow the prediction of future results. Typical applications include market segmentation, customer profiling, fraud detection, evaluation of retail

Changed: 2004-11-16 19:32:55 Changer: admin

Administration Menu

[Login] à [Main Menu] à [Administration]

This page was selected from the [Main Menu] via the {Administration} button. Only the SuperLuminate administrator (User ID =“admin”) has access to administration functions. From the [Administration] page select the button {Modify Configuration}.



Address: http://superluminate.com/sl/sl_main.php

[Administration] Access privileges are determined by your account role

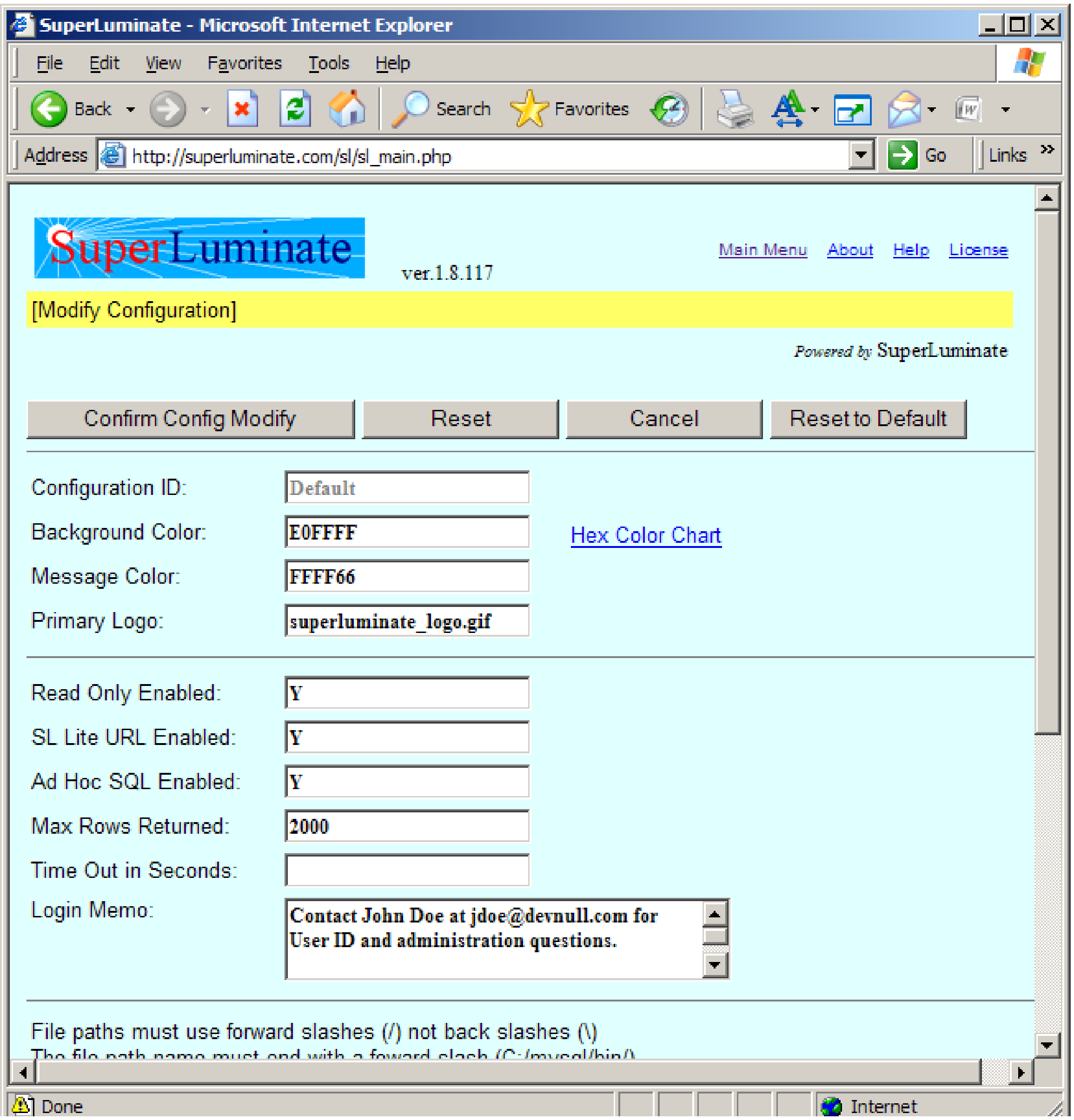
Modify Configuration Page

[Login] à [Main Menu] à [Administration] à [Modify Configuration]

Via this page your SuperLuminate administrator can change the look and feel of your SuperLuminate implementation including:

- Page color (Note: Hex Color Chart button displays sample colors)
- Message bar color
- Your organization logo, e.g. stanford_univ_gsb_logo.gif

For a complete list of fields and examples see “Administration Topics / System Configuration” in a later chapter of this guide.



http://superluminate.com/sl/sl_main.php

SuperLuminate ver.1.8.117

[Modify Configuration]

Configuration ID: Default

Background Color: E0FFFF

Message Color: FFFF66

Primary Logo: superluminate_logo.gif

Read Only Enabled: Y

SL Lite URL Enabled: Y

Ad Hoc SQL Enabled: Y

Max Rows Returned: 2000

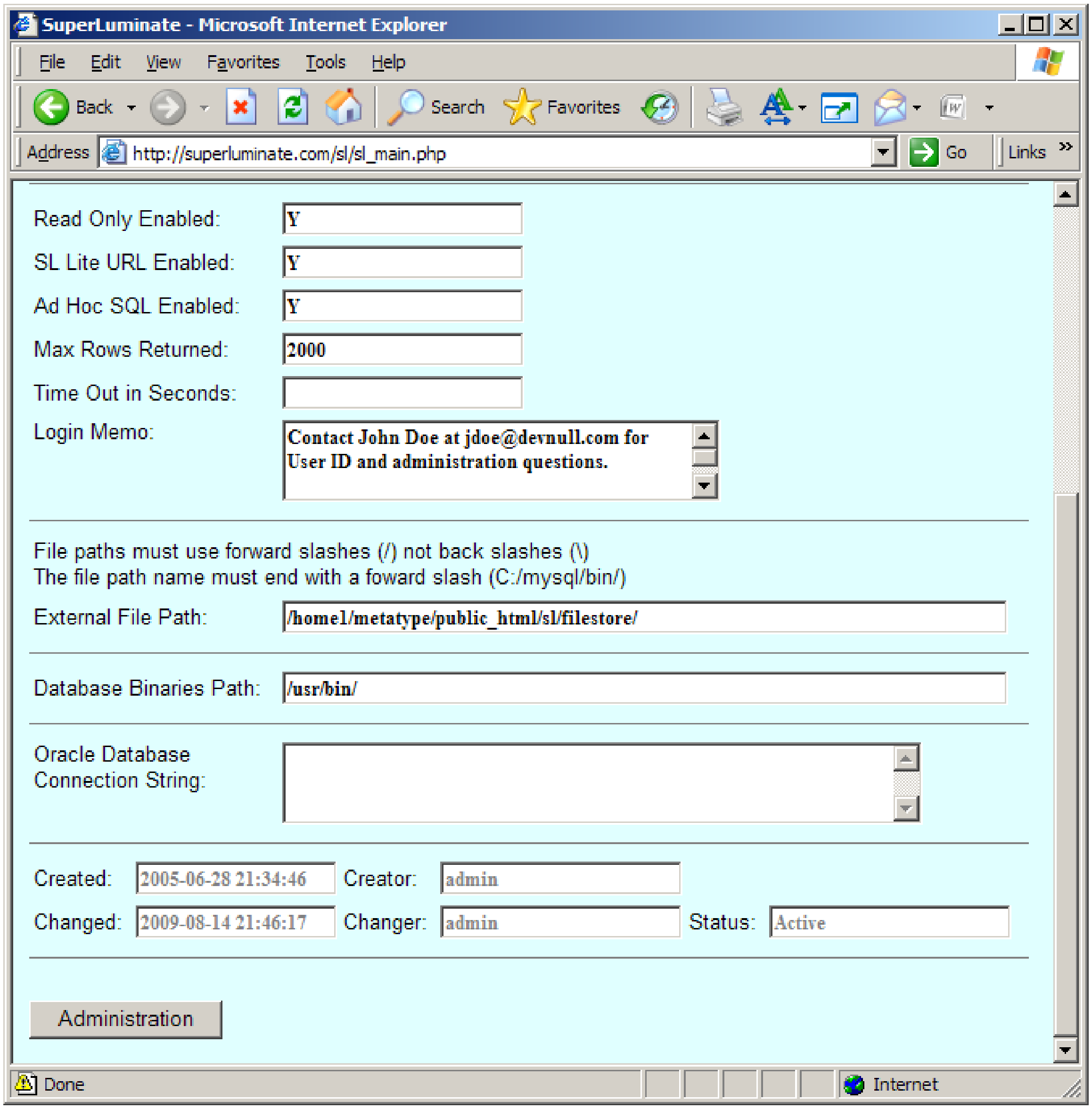
Time Out in Seconds:

Login Memo: Contact John Doe at jdoe@devnull.com for User ID and administration questions.

File paths must use forward slashes (/) not back slashes (\)

The file path name must end with a forward slash (C:/mysql/bin/)

And the lower portion of the configuration page:


SuperLuminate - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address
Go
Links
Administration
Done
Internet

http://superluminate.com/sl/sl_main.php

Read Only Enabled: Y

SL Lite URL Enabled: Y

Ad Hoc SQL Enabled: Y

Max Rows Returned: 2000

Time Out in Seconds:

Login Memo: Contact John Doe at jdoe@devnull.com for User ID and administration questions.

File paths must use forward slashes (/) not back slashes (\)
The file path name must end with a foward slash (C:/mysql/bin/)

External File Path: /home1/metatype/public_html/sl/filestore/

Database Binaries Path: /usr/bin/

Oracle Database Connection String:

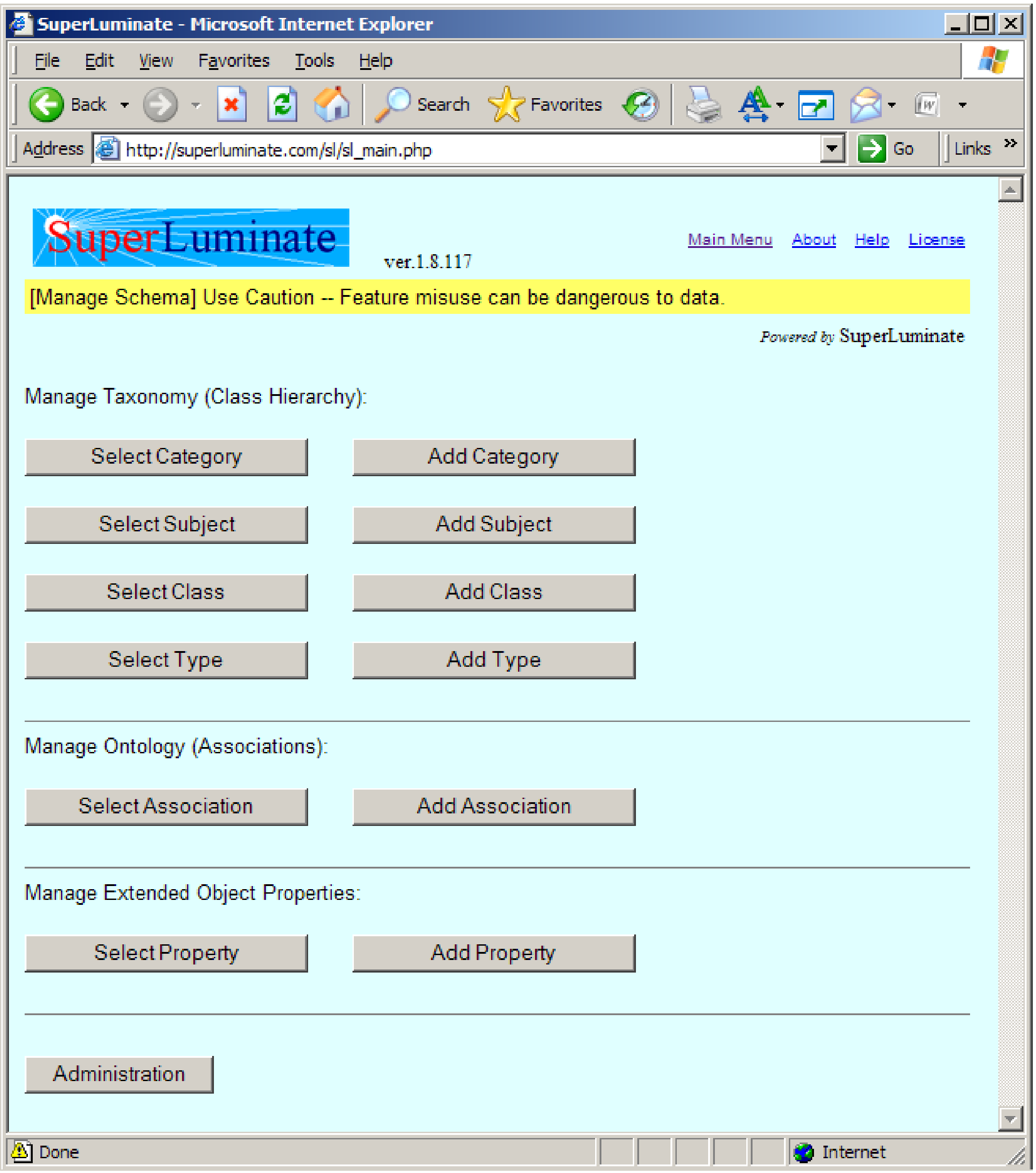
Created: 2005-06-28 21:34:46 Creator: admin

Changed: 2009-08-14 21:46:17 Changer: admin Status: Active

Manage Schema Menu

[Login] à [Main Menu] à [Administration] à [Manage Schema]

This page can be used by your SuperLuminate administrator to customize your installation by adding or customizing the Taxonomy, Schema (A.K.A. Information Model), and Extended Properties.



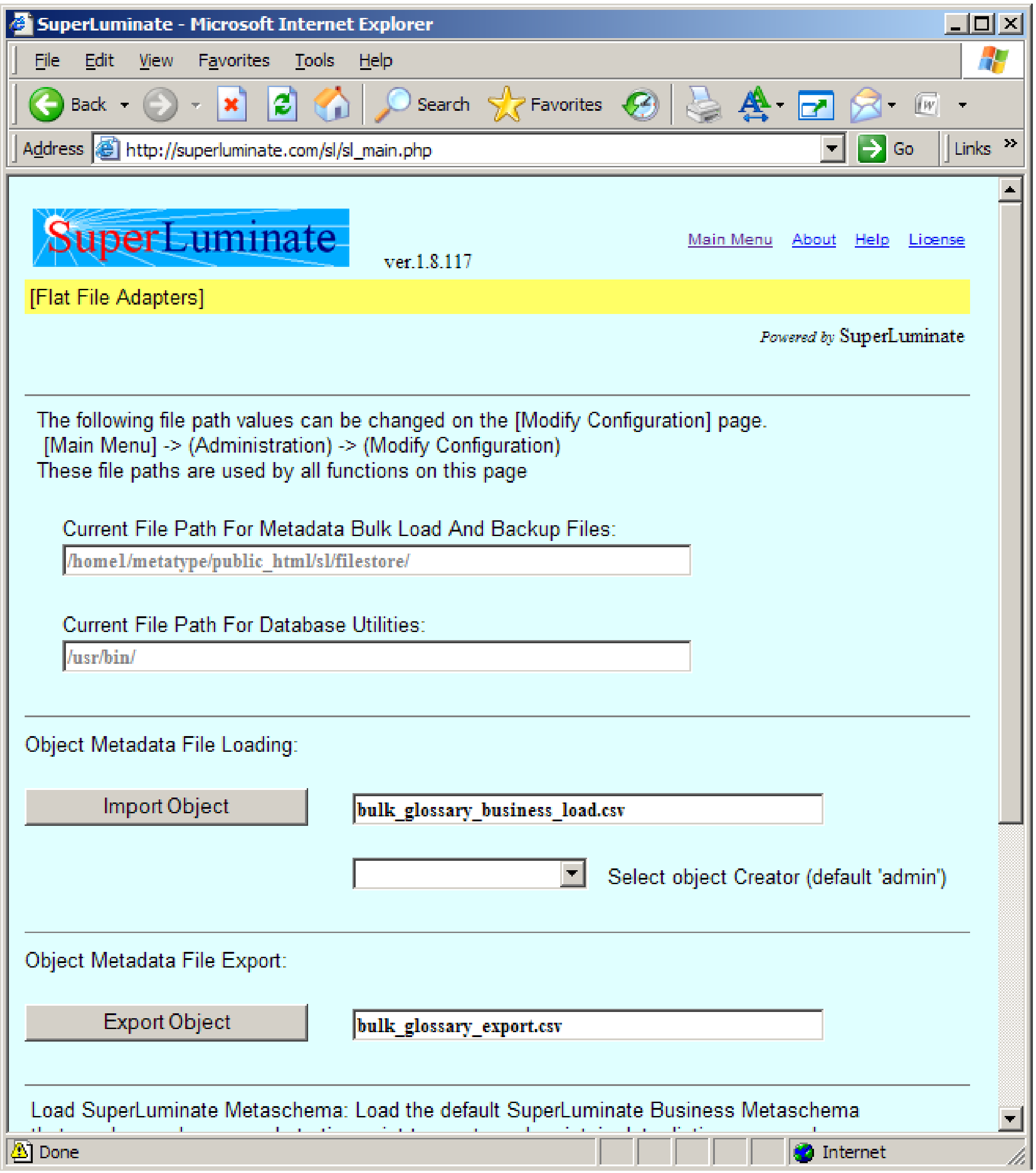
[Manage Schema] Use Caution -- Feature misuse can be dangerous to data.

Chang to User Guide

Flat File Adapters

[Login] à [Main Menu] à [Administration] à [Flat File Adapters]

This page can be used by your SuperLuminate administrator to backup the database and restore the entire data dictionary if required. The administrator can also import and export data from and to comma delimited text files (CSV).


SuperLuminate - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://superluminate.com/sl/sl_main.php Go Links
SuperLuminate
ver.1.8.117
Main Menu About Help License
[Flat File Adapters]
Powered by SuperLuminate
Import Object
Export Object
Done
Internet

The following file path values can be changed on the [Modify Configuration] page.
[Main Menu] -> (Administration) -> (Modify Configuration)
These file paths are used by all functions on this page

Current File Path For Metadata Bulk Load And Backup Files: /home1/metatype/public_html/sl/filestore/

Current File Path For Database Utilities: /usr/bin/

Object Metadata File Loading:

Import Object bulk_glossary_business_load.csv

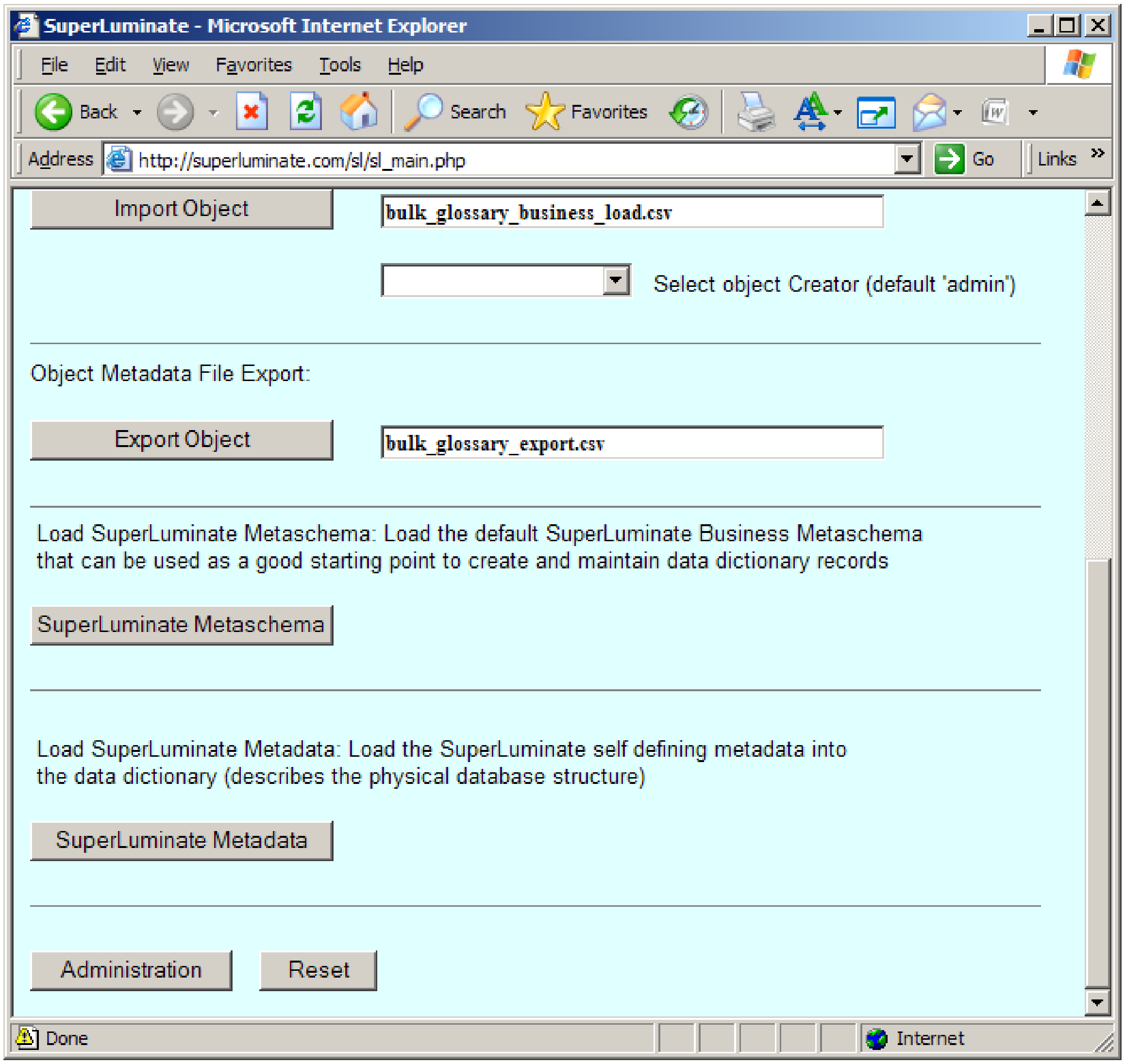
Select object Creator (default 'admin')

Object Metadata File Export:

Export Object bulk_glossary_export.csv

Load SuperLuminate Metaschema: Load the default SuperLuminate Business Metaschema

And the lower portion of the flat file adapters page:



SuperLuminate - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://superluminate.com/sl/sl_main.php Go Links

bulk_glossary_business_load.csv

Select object Creator (default 'admin')

Object Metadata File Export:

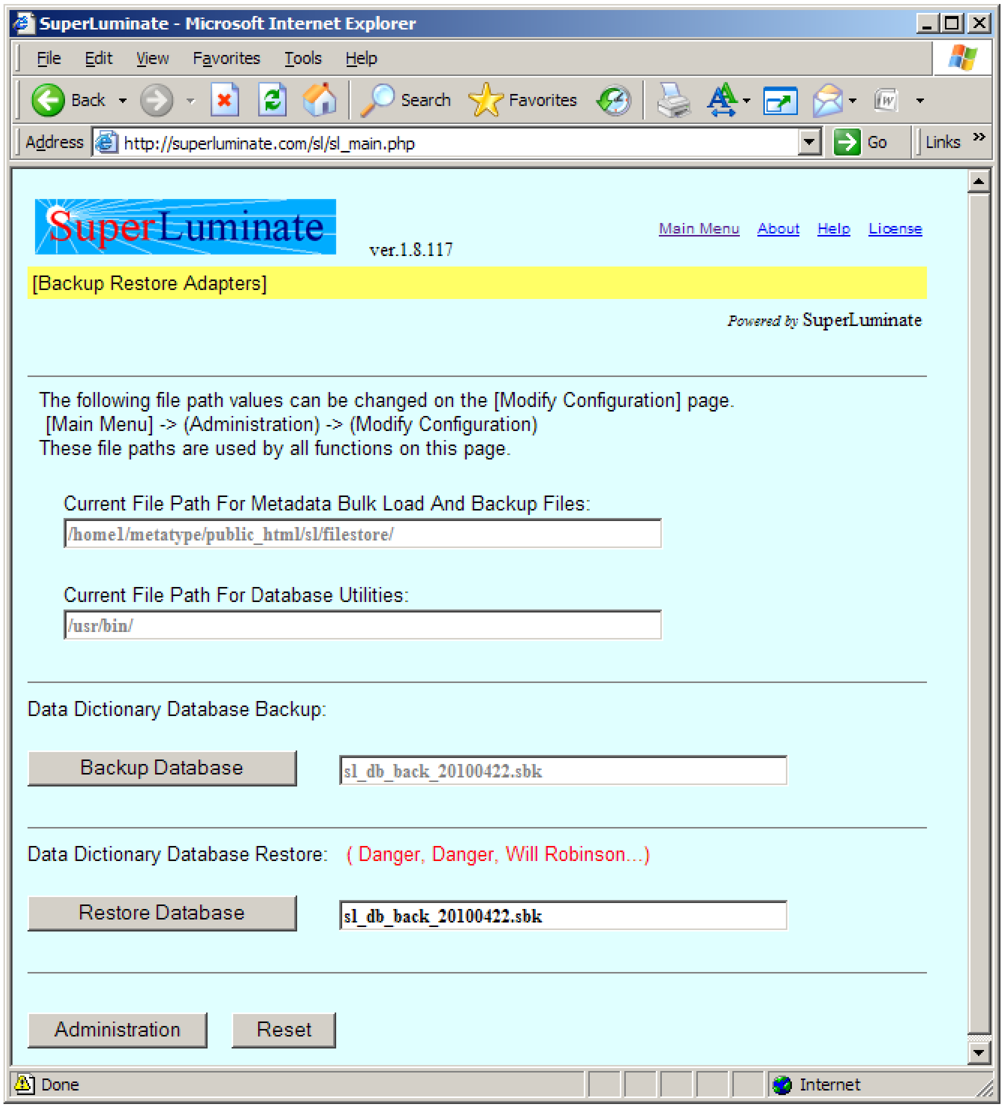
bulk_glossary_export.csv

Load SuperLuminate Metaschema: Load the default SuperLuminate Business Metaschema that can be used as a good starting point to create and maintain data dictionary records

Load SuperLuminate Metadata: Load the SuperLuminate self defining metadata into the data dictionary (describes the physical database structure)

Backup Restore Adapters

[Login] à [Main Menu] à [Administration] à [Backup Restore Adapters]


SuperLuminate - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://superluminate.com/sl/sl_main.php Go Links
SuperLuminate
ver.1.8.117
Main Menu About Help License
Powered by SuperLuminate
Backup Database
Restore Database
Administration
Reset
Done
Internet

[Backup Restore Adapters]

The following file path values can be changed on the [Modify Configuration] page.
[Main Menu] -> (Administration) -> (Modify Configuration)
These file paths are used by all functions on this page.

Current File Path For Metadata Bulk Load And Backup Files: /home1/metatype/public_html/sl/filestore/

Current File Path For Database Utilities: /usr/bin/

Data Dictionary Database Backup: sl_db_back_20100422.sbk

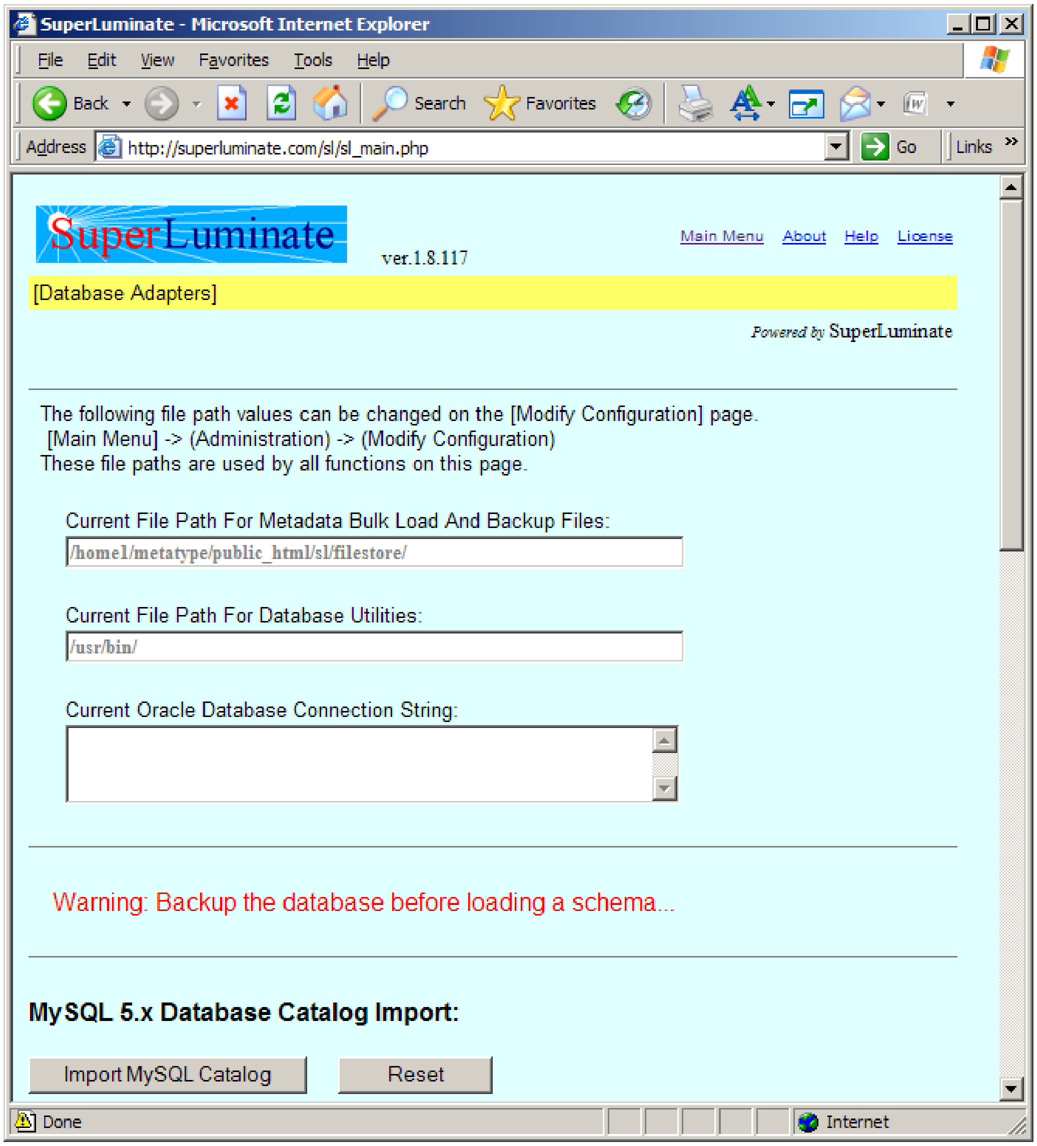
Data Dictionary Database Restore: (Danger, Danger, Will Robinson...) sl_db_back_20100422.sbk

Database Adapters

[Login] à [Main Menu] à [Administration] à [Database Adapters]

The following functions load (import) data from external database catalogs in the SuperLuminate data dictionary. The database catalog data will contain database, schema, table, and column physical definitions. Any notes or comments added to the database catalog by the database administrator will also be loaded. Loading the catalog metadata into SuperLuminate will enable the business to search for existing data resources and maintain business definitions and connections to other data dictionary objects including data models and reports.

Taxonomy Assignment for Catalog Loads: All loaded database catalog records will be assigned a temporary Catalog, Subject and Type name 'MySQL', 'Catalog', and 'Prod' respectively. These names can be changed via the [Manage Schema] page when you are satisfied with the load. The records will be assigned to Class 'Database', 'Table', and 'Column'.


SuperLuminate - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://superluminate.com/sl/sl_main.php Go Links
SuperLuminate
ver.1.8.117
Main Menu About Help License
Powered by SuperLuminate
Import MySQL Catalog
Reset
Done
Internet

[Database Adapters]

The following file path values can be changed on the [Modify Configuration] page.
[Main Menu] -> (Administration) -> (Modify Configuration)
These file paths are used by all functions on this page.

Current File Path For Metadata Bulk Load And Backup Files:
/home1/metatype/public_html/sl/filestore/

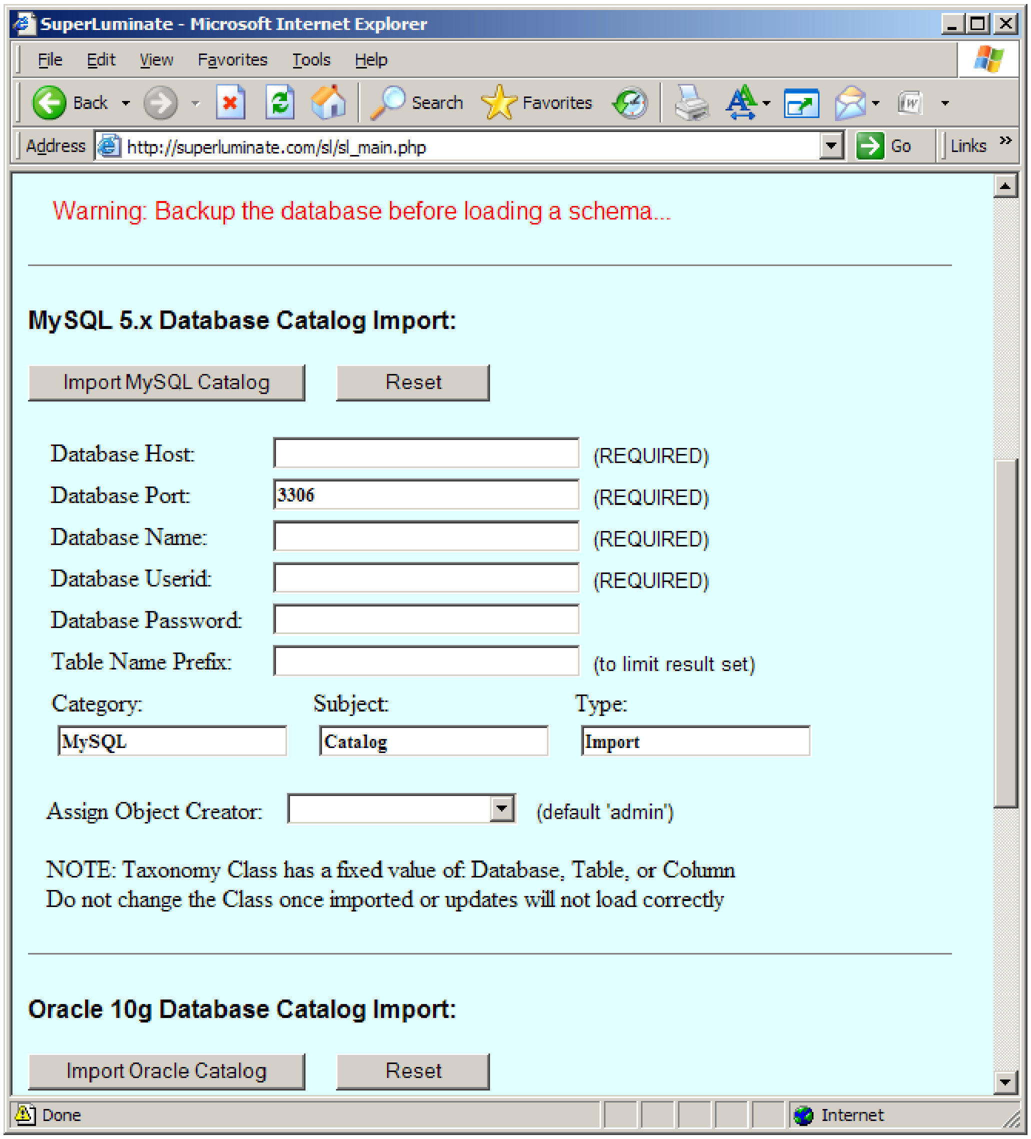
Current File Path For Database Utilities:
/usr/bin/

Current Oracle Database Connection String:

Warning: Backup the database before loading a schema...

MySQL 5.x Database Catalog Import:

When the schema that is being loaded is not already in the SuperLuminate data dictionary the load screen will indicate INSERT and all data will be loaded.



http://superluminate.com/sl/sl_main.php

Warning: Backup the database before loading a schema...

MySQL 5.x Database Catalog Import:

Database Host: (REQUIRED)
Database Port: 3306 (REQUIRED)
Database Name: (REQUIRED)
Database Userid: (REQUIRED)
Database Password:
Table Name Prefix: (to limit result set)

Category: MySQL
Subject: Catalog
Type: Import

Assign Object Creator: (default 'admin')

NOTE: Taxonomy Class has a fixed value of: Database, Table, or Column
Do not change the Class once imported or updates will not load correctly

Oracle 10g Database Catalog Import:

Example: To load the SL_Prod MySQL schema into SuperLuminate:

Database Host:	localhost
Database Port:	3306
Database Name:	SL_Prod
Database Userid:	root
Database Password:	WebPa55

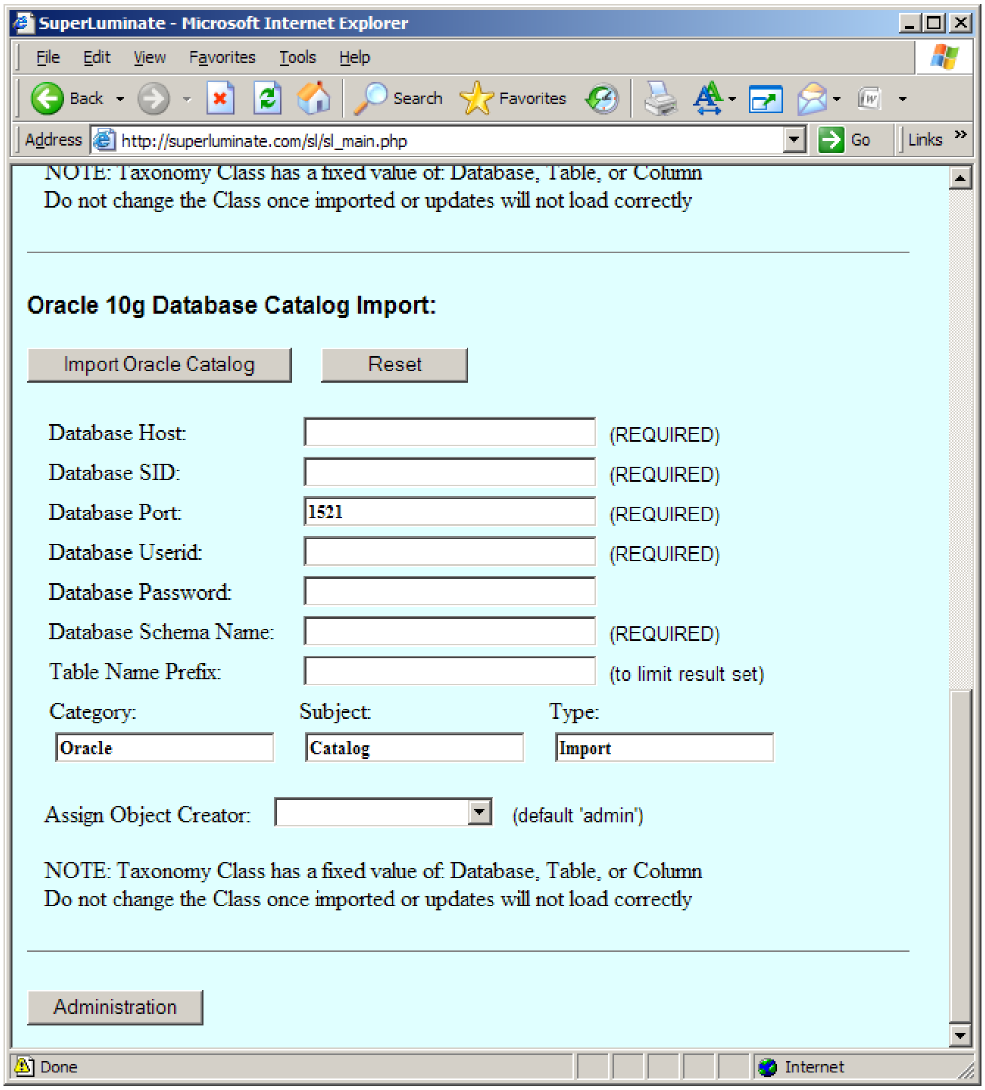
And hit the button (Import MySQL Catalog)

Table Name Prefix is null so all tables in the database will be selected. Assign Object Creator is null so “Admin” will be assigned as the record owner.

The Catalog, Subject, and Type fields are defaulted. They can be changed to anything you want. The taxonomy field values are used when adding the records to SuperLuminate. They do not need to previously exist but they can be used to search and select records once loaded.

For UNIX / Linux users remember (SL_Prod is not equal to sl_prod) UNIX is case sensitive – unlike Windows.

If the schema was load previously the load screen will indicate UPDATE. In update mode all records in the schema will be processed. If the record is already in the SuperLuminate data dictionary the comment from the source database record, changer, and changed date will be update on every record. Any new records will be inserted. NOTE: you can determine if a record was deleted from the source system by scanning all updated records – any record where the change date was not update means the corresponding record in the source system was deleted. SuperLuminate does not automatically delete the record because the user may have documentation for the record they wish to save.



http://superluminate.com/sl/sl_main.php

NOTE: Taxonomy Class has a fixed value of: Database, Table, or Column
Do not change the Class once imported or updates will not load correctly

Oracle 10g Database Catalog Import:

Database Host: (REQUIRED)
Database SID: (REQUIRED)
Database Port: 1521 (REQUIRED)
Database Userid: (REQUIRED)
Database Password:
Database Schema Name: (REQUIRED)
Table Name Prefix: (to limit result set)

Category: Oracle
Subject: Catalog
Type: Import

Assign Object Creator: (default 'admin')

NOTE: Taxonomy Class has a fixed value of: Database, Table, or Column
Do not change the Class once imported or updates will not load correctly

This processing method allows the user to update the database records with extensive “DESCRIPTIONS” and “USAGE” documentation. When the records are updated only the “COMMENT” field is updated from the source systems retaining the users information.

Page Hierarchy

Navigation starts at the [Login] page and progresses from page to page traversing the page hierarchy from top to bottom from left to right. Each menu or page is identified in the message bar with a set of square brackets, e.g. [Main Menu].

Note Again: Anything that is in square brackets, e.g. [Login] is the name of the page. Anything that is in wavy brackets, e.g. {User Login} is the name of an item that can be selected (pressed).

The following is the hierarchy of pages in the SuperLuminate application.

- [Login]
  - [Main Menu]
    - [Search]
      - [Search Result]
        - [Selection]
          - [Modify]
          - [Delete]
          - [Parent]
          - [Child]
          - [Version]
          - [Report]
          - [Copy]
    - [New Entry]
      - [Add]
    - [Reporting]
      - [Report Result]
      - [Query Result]
    - [Your Account]
      - [Modify Your Profile]
      - [Modify Your Password]
    - [Administration]
      - [Manage User]
        - [Admin Add User]
        - [Admin Select User]
          - [Admin Modify Profile]
        - [Admin Select Password]
          - [Admin Modify Password]

- [Management Reporting]
    - [Manage System Report]
- [Flat File Adapters]
- [Backup Restore Adapters]
- [Database Adapters]
- [Manage Schema]
    - [Search Category]
    - [Add Category]
    - [Search Subject]
    - [Add Subject]
    - [Search Class]
    - [Add Class]
        - [Select Class Category]
            - [Select Class Subject]
    - [Search Type]
    - [Add Type]
        - [Select Type Category]
            - [Select Type Subject]
                - [Select Type Class]
    - [Search Association]
    - [Add Association]
        - [Select Parent Category]
            - [Select Parent Subject]
                - [Select Parent Class]
                    - [Select Child Class]
    - [Search Property]
    - [Add Property]
        - [Select Property Category]
            - [Select Property Subject]
                - [Select Property Class]
- [Modify Configuration]

{Logoff}

Application Architecture

Note: This chapter serves as an outline that will be used to document the use and usage of each registered user related function.

Security

Security is organized into three levels: Read Only access, User access, and Administrator (admin) access.

Read Only users (external persons can gain access to the SuperLuminate system without a user id and password if read only access was enabled by the administrator via the configuration page) can read records that have the group level classification of “General” via the {Search} and {Reporting} main menu functions. If the administrator enabled SuperLuminate Lite (web based Glossary access) users when using SuperLuminate Lite will be logged in the system as a Read Only user.

Users (persons with an assigned user id and password) can read records that have the group level classification of “General” via the {Search} and {Reporting) main menu functions. Users can also create records via the {New Entry} function and maintain their account information via the {Your Account} function on the [Main Menu] page. Any record the User has created can be updated, versioned and or deleted by the User (the creating user “owns” the record). Only the owner of the record or the administrator can update a record. Other authorized users can {Copy} or {Version} a record making them the owner of that copy or version.

Administration users have access to all functions in SuperLuminate. An Administration user can create, read, change, and or delete any control or metadata in the system.

Logging into SuperLuminate

Session Management: When any type of user logs into SuperLuminate a session key is assigned to the user that is used throughout the session to enables access to the system until the user logs off. If the user does not physically log off via the {Logoff} button the user’s session key will automatically expire at midnight.

Read Only Access

If enabled by the administrator anyone can access SuperLuminate via the {Read Only Access} button on the [Login] page. A read only user can only access authorized records in the system and if enabled by the administrator can run some reports.

Write Access (Create, Update, Delete)

Registered users (users that have a user id and password to login) can create records and update the records they own.

Administration

The administration account is the only account that has full access to all functions and data in the system.

Group Level (General, Private)

Each record is assigned a group level. “General” group level records can be read by any read-only or registered user. Records that are classified with a group level of “Private” can only be read by the records owner or the administrator.

(This code will be used in future to enable group level user access across user groups.)

Status (Active, Inactive)

Each record is assigned a status. A record with a status of “Active” can be read by any read only or registered user. Records that are classified with a status of “Inactive” can only be read by the records owner or the administrator.

Session Logging (Audit)

Any user (read only, registered, administrator) that logs into SuperLuminate is recorded into a control table for auditing purposes. Your SuperLuminate administrator can access the audit reports via the [Management Reporting] page.

Search and Retrieval

Basic Search

By way of the [Search] page via the {Search} button on the [Main Menu] page any one or combination of the following fields can be used to search for information is SuperLuminate.

- Key Properties (Physical Database Names)
  - Key properties are those attribute that are used to uniquely identify any object record in SuperLuminate.
    - Category_TX e.g. Company Name
    - Subject_TX e.g. Department
    - Class_TX e.g. Glossary
    - Type_TX e.g. Finance
    - Name_TX e.g. Total Cost of Sales
    - Version_DC e.g. 1.000

- Base Properties (Physical Database Names)
  - Base properties are those attributes that are used to further describe any object record in SuperLuminate.
    - Alias_TX e.g. TCS

Desc_TX	e.g. Total cost of sales is the total production cost plus administration overhead, research and development overhead, and selling and distribution overhead.
Usage_TX	e.g. Used by Accounting and Finance.
Comment_TX	e.g. Maintained by John Doe in Accounting

- Control Properties (Physical Database Names)
  - Control properties are those attribute that are used to track and control any object record in SuperLuminate.

Creator_ID	e.g. admin
Created_DS	e.g. 2009-05-14 17:05:22
Changer_ID	e.g. admin
Changed_DS	e.g. 2009-05-14 17:05:22
Group_CD	e.g. General
Status_CD	e.g. Active

Advanced Search

- Extended Properties
  - Extended properties are those attribute that have been added by the administrator to any specific class of objects in SuperLuminate. Extended properties are custom properties that can extend the number of attributes for an object, e.g. The administrator could have added the following Extended Properties for objects of Class "Column":

Column Datatype	e.g. varchar
Column Size	e.g. 2000
Column Can Be Null	e.g. yes
Column Sequence	e.g. 10

Relationship Navigation

Relationship navigation linkages are maintained to allow a user to navigate up, down and across from one definition to another and back again. A common example is: A database has many tables and each table has many columns. If you are displaying the definition of a table in on the [Selection] page you will see toward the bottom of the

[Selection] page a link to the parent record “database’ and a child linkage for each of the “columns” associated with the table being displayed.

- Parent Relationships
  After a successful search an object record is displayed on the [Selection] page. Towards the bottom of the page is the Parent Objects section of the object definition. If the object being displayed is a child to another object via a relationship linkage. Links to the parent or parent objects records will be displayed.

- Child Relationships
  After a successful search an object record is displayed on the [Selection] page. Towards the bottom of the page is the Child Objects section of the object definition. If the object being displayed is the parent of another object via a relationship linkage. Links to the child or children objects records will be displayed.

Reporting

- Canned Reports
  Several canned reports are available via the {Reporting} button on the [Main Menu] page.

- Ad-Hoc Reports
  Ad hoc reports can be run in the Ad Hoc SQL window via the {Reporting} button on the [Main Menu] page. Read Only users cannot run ad hoc reports. If the administrator has enabled ad hoc reports for registered users, the user can create and run SQL selection statements via the AD Hoc SQL window. The administrator can run any SQL via the Ad Hoc SQL window.

- Administration Reports
  Several canned reports are available on the [Management Reporting] page.
  [Main Menu] à [Administration] à [Management Reporting]

Global Variables

Global variables are used to set site specific values, e.g. database connection values. Global variables are contained in the SuperLuminate file sl_globals.php

```
// Connection variables (Set for your site)
$g_host                 = "localhost";          // Server URL or IP Address
```

```
$g_user          = "root";          // Database Super User Login
$g_pass          = "WebPa55";       // Database Super User Password
$g_database      = "SL_Prod";       // Database Name

// Configuration variables
$g_main          = "sl_main.php";   // Main SuperLuminate application
$g_version       = "ver.1.8.117";   // SuperLuminate Version
```

Functional Architecture

The SuperLuminate application (sl_main.php) is designed as a monolithic block. Most of the code is in one file. The SuperLuminate application is a classic "function-main" call structure. From logon to logoff, all functionality and navigation takes place by way of a set of function calls.

Functional Segmentation

For ease of maintenance the SuperLuminate application is broken into distinct sections. Each section has a specific function.

Function Calling Protocol

The primary variable used in SuperLuminate for navigation is $auggie. The value of Auggie determines what function to call (execute).

```
// *********************************************************************
// SECTION: Function Description
// *********************************************************************

// FUNCTION:                               AUGGIE:

// --- SECTION: Generic Functions

// GenerateHeader                          n/a
// GenerateHeaderLite                      n/a
// FormButtonsMainMenu                     Cancel
// FormButtonsMainMenu                     Main Menu

// --- SECTION: Main Menu

// FormSearchObject                        Search
// FormAddObject                           New Entry
// FormButtonsReporting                    Reporting
// FormButtonsUserAccount                  Your Account
// FormButtonsAdmin                        Administration
// FormLogin                               Logoff

// --- SECTION: Search

// FormSearchLite                          n/a
// ProcSearchLite                          Find Now
// FormDisplayLite                         preauggie=L-
// FormSearchLite                          Find Again

// ProcSearchObject                        Search Now
// FormSearchObject                        Search Again
// ProcBulkReport                          Bulk Report
// FormDisplayObject                      preauggie=O-

// --- SECTION: Display Object

// FormModifyObject                       Modify
// ProcModifyObject                       Confirm Object Modify

// FormCopyObject                         Copy
// ProcCopyObject                         Confirm Object Copy

// FormDeleteObject                       Delete
// ProcDeleteObject                       Confirm Delete

// FormObjectParentClass                  Parent
// FormAddObjectParent                    Confirm Parent Class
// ProcAddObjectParent                    Confirm Add Parent

// FormObjectChildClass                   Child
// FormAddObjectChild                     Confirm Child Class
// ProcAddObjectChild                     Confirm Add Child

// ProcVersionObject                      Version

// FormDeleteRelationship                 preauggie=R-
// ProcDeleteRelationship                 Confirm Delete Relationship

// --- SECTION: New Entry

// FormAddObjectProperties                Confirm Object Add
// ProcAddObject                          Insert New Object

// --- SECTION: Your Account

// FormModifyYourProfile                  Modify Your Profile
// ProcModifyYourProfile                  Confirm Profile Change

// FormModifyYourPassword                 Modify Your Password
// ProcModifyYourPassword                 Confirm Password Change

// --- SECTION: Administration

// FormButtonsManageUser                  Manage User
// FormButtonsManagementReporting         Management Reporting
// FormButtonsFlatFileAdapters            Flat File Adapters
// FormButtonsBackupRestoreAdapters       Backup Restore Adapters
// FormButtonsDatabaseAdapters            Database Adapters
// FormButtonsManageSchema                Manage Schema
// FormModifyConfiguration                Modify Configuration

// --- SECTION: Modify Configuration

// ProcModifyConfig                       Confirm Config Modify
// ProcModifyConfig                       Reset To Default

// --- SECTION: Manage User

// FormAddUser                            Add User
// ProcAddUser                          Confirm User Add

// FormSelectUserProfile                Select Profile to Modify
// FormModifyUserProfile                Modify User Profile
// ProcModifyUserProfile                Confirm User Change

// FormSelectUserPassword               Select Password to Modify
// FormModifyUserPassword               Modify User Password
// ProcModifyUserPassword               Confirm User Password Change

// FormModifyObjectCreator              Modify Object Creator
// ProcModifyObjectCreator              Confirm Creator Modify

// --- SECTION: Manage Schema

// FormAddCategory                      Add Category
// ProcAddCategory                      Confirm Category Add

// ProcSelectCategory                   Select Category
// FormDeleteSchemaCategory             preauggie=X-
// ProcDeleteSchemaCategory             Confirm Category Delete
// FormModifySchemaCategory             preauggie=K-
// ProcModifySchemaCategory             Confirm Category Modify

// FormAddSubject                       Add Subject
// ProcAddSubject                       Confirm Subject Add

// ProcSelectSubject                    Select Subject
// FormDeleteSchemaSubject              preauggie=J-
// ProcDeleteSchemaSubject              Confirm Subject Delete
// FormModifySchemaSubject              preauggie=I-
// ProcModifySchemaSubject              Confirm Subject Modify

// FormSelectClassCategory              Add Class
// FormSelectClassSubject               Select Class Subject
// FormAddClass                         Enter New Class
// ProcAddClass                         Confirm Class Add

// ProcSelectClass                      Select Class
// FormDeleteSchemaClass                preauggie=H-
// ProcDeleteSchemaClass                Confirm Class Delete
// FormModifySchemaClass                preauggie=G-
// ProcModifySchemaClass                Confirm Class Modify

// FormSelectTypeCategory               Add Type
// FormSelectTypeSubject                Select Type Subject
// FormSelectTypeClass                  Select Type Class
// FormAddType                          Enter New Type
// ProcAddType                          Confirm Type Add

// ProcSelectType                       Select Type
// FormDeleteSchemaType                 preauggie=F-
// ProcDeleteSchemaType                 Confirm Delete Type
// FormModifySchemaType                 preauggie=E-
// ProcModifySchemaType                 Confirm Modify Type

// FormSelectParentCategory             Add Association
// FormSelectParentSubject                Select Parent Subject
// FormSelectParentClass                  Select Parent Class
// FormAddAssociation                     Select Child Class
// ProcAddAssociation                     Confirm Association Add

// ProcSelectAssociation                  Select Association
// FormDeleteSchemaAssoc                  preauggie=D-
// ProcDeleteSchemaAssoc                  Confirm Association Delete
// FormModifySchemaAssoc                  <removed>
// ProcModifySchemaAssoc                  <removed>

// FormSelectPropCategory                 Add Property
// FormSelectPropSubject                  Select Property Subject
// FormSelectPropClass                    Select Property Class
// FormAddProperty                        Enter New Property
// ProcAddProperty                        Confirm Property Add

// ProcSelectProperty                     Select Property
// FormDeleteSchemaProp                   preauggie=B-
// ProcDeleteSchemaProp                   Confirm Property Delete
// FormModifySchemaProp                   preauggie=A-
// ProcModifySchemaProp                   Confirm Property Modify

// --- SECTION: Utility Functions

// ProcErrMsg                             n/a
// ProcLogEvent                           n/a
// ProcSearchDropDown                     n/a
// ProcAdHocQuery                         Execute Query
// ProcSystemSettings                     System Settings
// ProcCallDatabase                       n/a
// ProcCallDatabaseMySQL                  n/a
// ProcCallDatabaseOracle                 n/a
// ProcSelectBox                          n/a
// ProcLoadMetadata                       SuperLuminate Metadata
// ProcImportMySQLCatalog                 Import MySQL Catalog
// ProcImportOracleCatalog                Import Oracle Catalog
// ProcImportObject                       Import Object
// ProcExportObject                       Export Object
// ProcBackupDatabase                     Backup Database
// ProcRestoreDatabase                    Restore Database
// ProcSetupDatabase                      n/a
// ProcLoadMetaschemaActive               SuperLuminate Metaschema
// ProcLoadMetaschemaQuiet                n/a

// --- SECTION: Canned Query Functions

// ProcExecCannedQuery                          DB Processes
// ProcExecCannedQuery                          DB Status
// ProcExecCannedQuery                          DB Variables
// ProcExecCannedQuery                          Event Log Today
// ProcExecCannedQuery                          Event Log One Week
// ProcExecCannedQuery                          Event Log One Month
// ProcExecCannedQuery                          Category Count
// ProcExecCannedQuery                          Subject Count
// ProcExecCannedQuery                          Class Count
// ProcExecCannedQuery                          Type Count
// ProcExecCannedQuery                          Creator Count
// ProcExecCannedQuery                          Changer Count
// ProcExecCannedQuery                          Group Count
// ProcExecCannedQuery                          Status Count
// ProcExecCannedQuery                          Object Count
// ProcExecCannedQuery                          Relationship Count
// ProcExecCannedQuery                          Class Hierarchy
// ProcExecCannedQuery                          Association List
// ProcExecCannedQuery                          Extended Props

// --- SECTION: JavaScript Functions

// JSopenReportWindow                           n/a
// JSopenAboutWindow                            n/a
// JSopenHelpWindow                             n/a
// JSopenLicenseWindow                          n/a
// JSopenColorChartWindow                       n/a

// *******************************************************************
// SECTION: Globals
// *******************************************************************
```

Conceptual Architecture

Three Schema Architecture

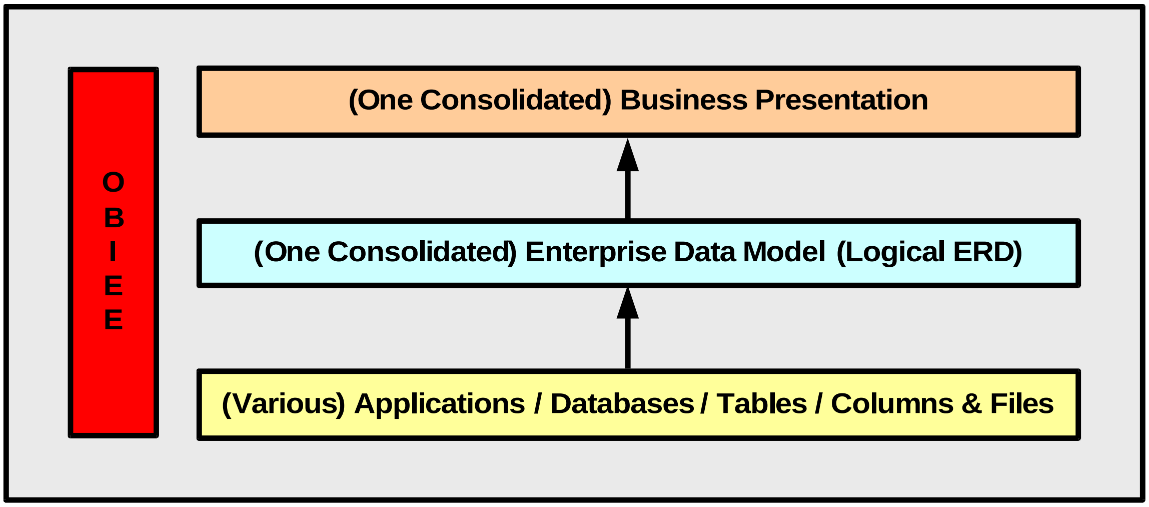
SuperLuminate is built on an industry accepted “Three Schema Architecture.”

Presentation Layer	A.K.A. External	or	Reporting
Logical Layer	A.K.A. Conceptual	or	Business Rules
Physical layer	A.K.A. Internal	or	System

With SuperLuminate you can define and maintain information (metadata) that maps and documents the complex relationships that are present in industry leading tools and applications, e.g. Oracle OBIEE, and Oracle BI Apps.

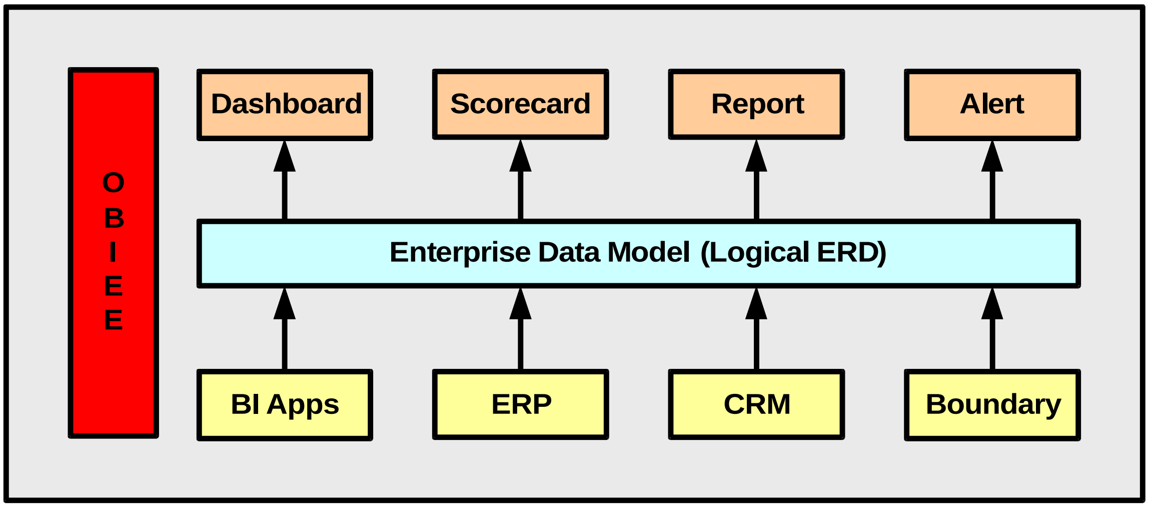
Oracle OBIEE (Oracle Business Intelligence Enterprise Edition) is an industry leading Business Intelligence (BI) reporting tool. By way of OBIEE a business end user can create a report or dashboard using various forms of physical data without knowing the format or location of the data itself. This concept is known as data abstraction. The mapping from physical to logical and then to the report (presentation layer) is contained in the OBIEE repository. The OBIEE repository is internal to OBIEE and does not provide a user interface to business level documentation.

SuperLuminate can be used in conjunction with the OBIEE repository to provide the business descriptions required to understand what has been implemented including existing dashboard and report definitions.


OBIEE
(One Consolidated) Business Presentation
(One Consolidated) Enterprise Data Model (Logical ERD)
(Various) Applications / Databases / Tables / Columns & Files

SuperLuminate can help you map and define the complex relationships and definitions that make up your Information Technology (IT) infrastructure. For example, Oracle sells an industry leading Data Warehousing application known as “BI Apps” (Business Intelligence Analytic Applications). BI Apps serves as a starting point for many organizations to building out their Enterprise Data Warehouse (EDW).

As you customize your out-of-the-box purchased application systems, SuperLuminate can help you create and maintain your out-of-the-box documentation and follow on customizations.



Conceptual Object Design

An object is the center piece of SuperLuminate metadata capture and maintenance. The object is what the user wants to store, maintain, and retrieve from the data dictionary. The power of SuperLuminate is based on the way SuperLuminate stores all information in a "Soup" – one place where all the "Apples & Oranges" are stored.

Abstracting all of the different kinds of metadata records into one generic object is the key to enabling common functionality, and extendibility. Extensions are made logically not physically. Extensions are added to SuperLuminate by adding or modifying control data to various control tables. These control tables define the schema (what the user can enter). In turn the control tables define the metaschema, the physical database structures maintained by SuperLuminate.

The columns in the database that capture the object definition are broken down into the following categories:

Object Key Properties

Every object can be uniquely identified by a combination of the following six attributes:

Category_TX	e.g. SuperLuminate

Subject_TX	e.g. Business
Class_TX	e.g. Glossary
Type_TX	e.g. Demo
Name_TX	e.g. Metadata
Version_DC	e.g. "1.000"

Object Base Properties

Every object has the following four additional attributes to segregate different forms of extended definition:

Alias_TX	e.g. metadata
Desc_TX	e.g. Metadata is data about data
Usage_TX	e.g. Maintained for finance
Comment_TX	e.g. Review for Q4 finals

Object Control Properties

Every object has the following set of control attributes to manage security and integrity:

Group_CD	e.g. General (used for access groups)
Source_ID	e.g. loader (not displayed to user)
Creator_ID	e.g. admin
Created_DS	e.g. 2003-06-14
Changer_ID	e.g. john
Changed_DS	e.g. 2004-08-20
Status_CD	e.g. Active

Object Extended Properties

Every object can have additional properties added. These properties are added by your SuperLuminate administrator. The values stored in the properties can then be maintained by the object owner.

Object Relationships

If allowed by the schema definition an object can be associated with one or more other objects.

[parent object class]	e.g. Table
[child object class]	e.g. Column

NOTE: Currently SuperLuminate only allows relationships between Class objects within the same schema. That is, Class objects that have the same Category and Subject defined. SuperLuminate does not allow intra-schema relationships – schema objects can only be associated to other objects in the same schema. This restriction ensures data integrity.

Schema Modification (A.K.A. Information Model)

The schema can be easily changed and maintained by the SuperLuminate administrator to meet the needs of your enterprise.

Taxonomy

The taxonomy enables the SuperLuminate administrator to define a high level standard system of classification for the organization. The SuperLuminate taxonomy comprises a four tier hierarchy from category to subject to class and finally to type. The four tier taxonomy was designed specifically to categorize Data Administration metadata.

Example:

For biological sciences, the taxonomy for classifying organisms, based mainly on physical similarities, comprises a hierarchy of seven primary levels. They are from top to bottom: Kingdom; Phylum; Class; Order; Family; Genus; and Species.

Entity Relationship Diagrams (ERD)

Schema(s) are attached to the Taxonomy at the Class level. A schema (A.K.A. Information Model) is a Class or set of Classes, that can have Extended Properties and may have Associations that define the relationships between Classes.

Note:

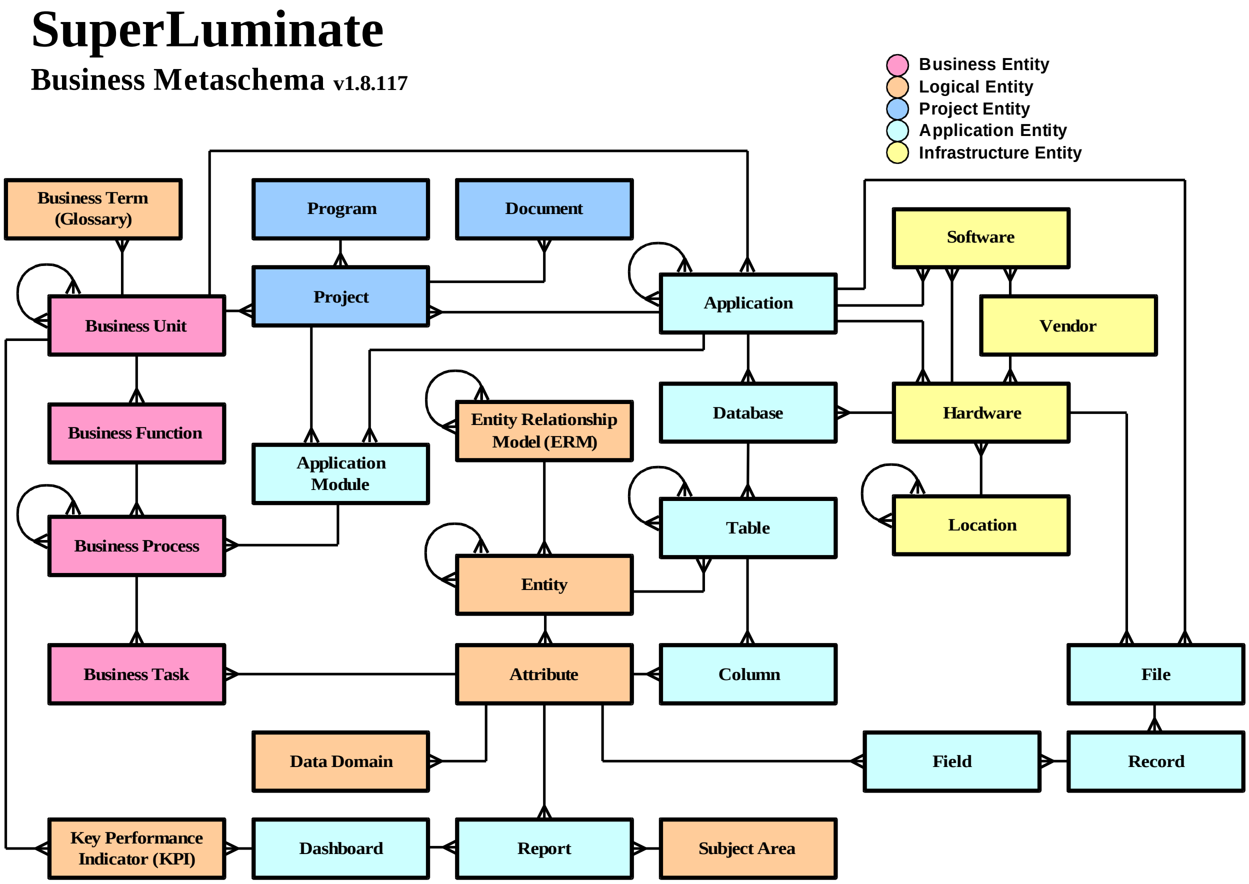
In a classical Information Model a Class would be called an Entity, and the Association would be called a Relationship, i.e. Entity Relationship Diagrams (ERD). This terminology is not used because of the nature of SuperLuminate in that it blends both ERD and object modeling methods.

Out-of-the-box (default) Schema and Taxonomy

The out-of-the-box solution is meant to serve as a starting point for your company but is in itself a fully functional implementation that will get you a long way.

SuperLuminate Business Metaschema (Metadata Structure)

This metaschema defines the underlying business metadata objects and their relationships that can be captured and maintained in the SuperLuminate data dictionary. This is a fully functional business metaschema but can easily be changed to meet the specific requirement of your enterprise.



Relationship Scenarios:

A relationship is a link between two entities, e.g. the entity “Table” is linked to the entity “Column” – “Table” is parent and “Column” is child.

An entity can also be related to itself, e.g. a “Table” entity can be a parent to many other “Table” entities and at the same time a “Table” entity can also be a child to many other “Table” entities.

In both cases the relationship is only a physical link maintained in the SuperLuminate database.

In some modeling methodologies a relationship is maintained in its own entity type (a relationship entity type) – the two entities (parent and child) must link separately to the relationship entity. The advantage to this is the relationship can have its own attribution (definition) including rules and options. The disadvantage is most relationships do not require attribution – they are only links. Creating a relationship entity type is cumbersome requiring a lot more maintenance. IRDS (ISO) metadata standards are averse to modeling relationships as entities.

Appropriate Taxonomy:

Category:	Enterprise
Subject:	Business
Class:	Entity
Type:	Opus

Data Object (Entity) Definitions:

- Application
  - Application software is computer software designed to help the user perform a particular task. Such programs are also called software applications, applications or apps.

- Application Module
  - An application can contain or consist of one or more application modules. An application can have no modules.

- Attribute
  - In computing, an attribute is a specification that defines a property of an object, element, or file. An attribute usually consists of a name, a data type, a size, and a default value.

- Business Function
  - A business function is a group of logically connected tasks performed together, in some logical sequence, to accomplish an objective or deliver a business product. An upper level business activity that is achieved via the performance of component activities. Examples: Manufacturing, Shipping.

- Business Process
  - A business process is a set of associated procedures or activities with defined roles and relationships carried out to realize a business function in pursuit of business objectives. A business process is the action taken to respond to particular events, convert inputs into outputs, and produce particular results. Business processes are what the enterprise must do to conduct its business successfully. A series of related business activities aimed at achieving one or more business objectives in a measurable manner.

- Business Task
  - A business task is a specific is a discrete event within a business processes. A business task will have only one input and one output.

- Business Term (Glossary)
  - A business term is the definition for common terminology and acronyms used in business plans, accounting, finance, and other aspects of the business.

- Business Unit
  - When companies become really large, they are best thought of as being composed of a number of businesses, i.e. business units. These organizational entities are large enough and homogeneous enough to exercise control over most strategic factors affecting their performance. They are managed as self contained planning units for which discrete business strategies can be developed.

- Column
  - In computing, a column is a specification that defines a property in a database table. A column usually consists of a name, a data type, a size, and a default value.

- Dashboard
  - Dashboard is a term now being used generally to refer to a web-based technology page on which real time information is collated from various sources in the business. The metaphor of dashboard is adopted here to emphasize the nature of the data being displayed on the page; it is a real-time analysis as to how a business is operating, just like on an automobile dashboard real time information is displayed about the performance of that vehicle.

- Data Domain
  - In data management and database analysis, a data domain refers to all the unique values which a data element may contain. The rule for determining the domain boundary may be as simple as a data type with enumerated list of values. For example, a database table that has information about people, with one record per person, might have a "gender" column. This gender column might be declared as a string data type, and allowed to have one of two known code values: "M" for male, "F" for female, or "U" for unknown. The data domain for the gender column is: "M", "F", and “U".

- Database
  - A Database is an integrated collection of logically related records consolidated into a common pool. A database stores detailed data needed to support the operations of an entire organization. Relational databases are the most commonly used database today. Relational databases use tables to structure information so that it can be readily and easily accessed and maintained.

- Document

  - A document is any file containing text, media or hyperlinks. A document (noun) is a bounded physical representation of a body of information designed with the capacity (and usually intent) to communicate.

- Entity
  - Entities can be thought of as nouns. An entity may be defined as a thing which is recognized as being capable of an independent existence and which can be uniquely identified. An entity may be a physical object such as a house or a car, an event such as a house sale or a car service, or a concept such as a customer transaction or order.

- Entity Relationship Model (ERM)
  - In software engineering, an entity-relationship model (ERM) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. Diagrams created by this process are called entity-relationship diagrams, ER diagrams, or ERDs.

- Field
  - In computing, a field is a specification that defines a property in a record. A field usually consists of a name, a data type, a size, and an integer number identifying the position of the field in the record.

- File
  - A file is a system of organizing data so that it can be understood by a particular application. A standardized file format makes it possible for different programs to share the same information. A file can contain one or more types of records as defined in the entity named “Record.”

- Hardware
  - Hardware is a general term that refers to the physical artifacts of a technology. Hardware are the physical components of a computer system. The physical or mechanical devices that comprise a computer system, such as the central processing unit, monitor, keyboard, mouse, other peripheral devices including telecommunications, and other information technology devices.

- Key Performance Indicator (KPI)
  - A key performance indicator is a high-level measurement of how well an organization is doing in achieving critical success factors. Distinguished from other metrics, key performance indicators (KPIs) are those metrics most critical to gauging progress toward objectives.

- Location

  - A location is a physical place identified by a postal address but the location specification can be much more specific, e.g. Building, Floor, Room, Cube, Chassis, Frame, Shelf, Rack.

- Program
  - A program or program management is the process of managing multiple interdependent projects that lead towards an improvement in an organization's performance.

- Project
  - A project in business is a collaborative enterprise, frequently involving research and design that is carefully planned to achieve a particular aim.

- Record
  - A record is a particular row in a file containing many fields. If there is only one type of record in a specific file then only one “Record” entity will exist for that file – one “Record” entity for each record type (format).

- Report
  - A report is a document characterized by information or other content reflective of inquiry or investigation, which is tailored to the context of a given situation and audience. The purpose of reports is usually to inform.

- Software
  - Computer software or just software is a general term used to describe a collection of computer programs (sets of instructions for a computer), procedures and documentation. There are two kinds of software: system software and application software. System software is usually stored on a computer's hard drive until needed by the computer. Application software ('apps' for short) are more commonly known as programs.

- Subject Area
  - An area of major interest or importance to the enterprise. It is centered on a major resource, product, or activity. The subject areas provide reference information when conducting detailed requirements gathering.

- Vendor
  - A vendor is a seller – someone who promotes or exchanges goods or services for money. A vendor, or a supplier, is a supply chain management term meaning anyone who provides goods or services to a company.

- Table
  - A table is a tabular view of data, in a relational database management system, defined by one or more columns of data and a primary key. A table is populated by rows of data.

- Table Association
    - A table association records the relationships (connections) between one or more tables.

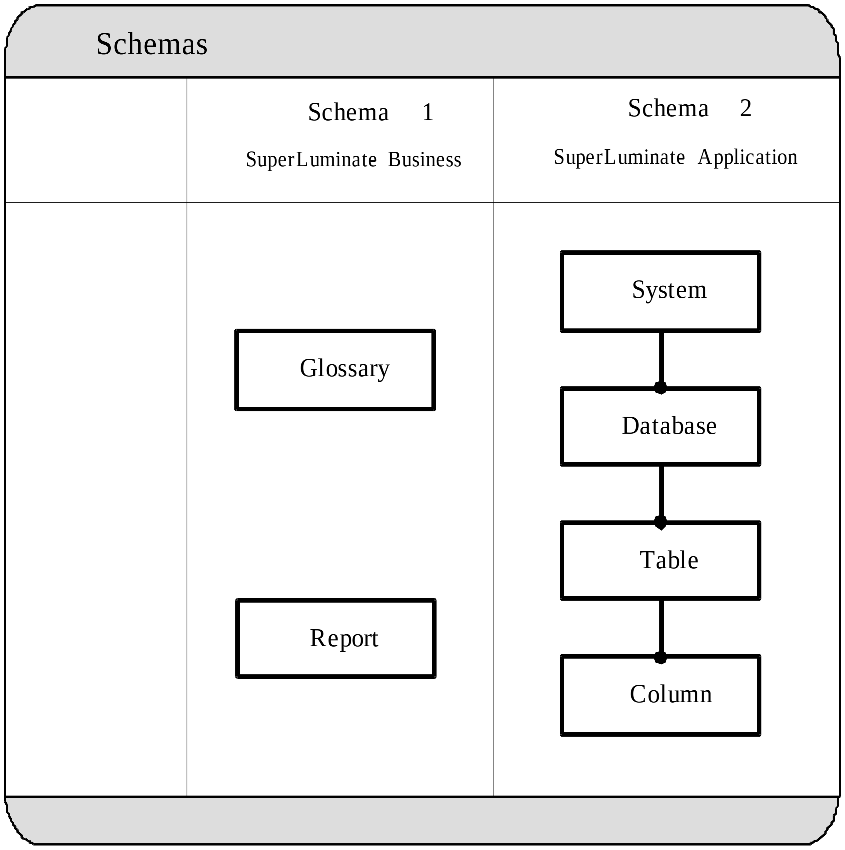
Extended Attribution:

Once the Metaschema is developed showing the relationships between all of the entities (and agreement is reached to the structure) the entities must be attributed. The required attributes must be added to each entity to record the required and optional information for each object.

Schemas:

SuperLuminate supports only one taxonomy but can support many schemas. Out-of-the-box SuperLuminate contains one schema. Schemas are defined at the subject level – If SuperLuminate contains two distinct Subjects it contains two schemas. A set of classes are defined for the schema in the SL_Class table. The Associations between the classes in the schemas are defined in the SL_Assoc table.

Schemas are defined by Category and Subject pairs, the associated classes and the Relationships between Classes. Classes are associated with one Category and Subject pair. If the class Report existed in more than one Category and Subject pair it would be a member in multiple schemas by name only. A Class can only exist in one schema – classes cannot span schemas. Relationships can only associate classes within the same schema – relationships cannot span schemas.



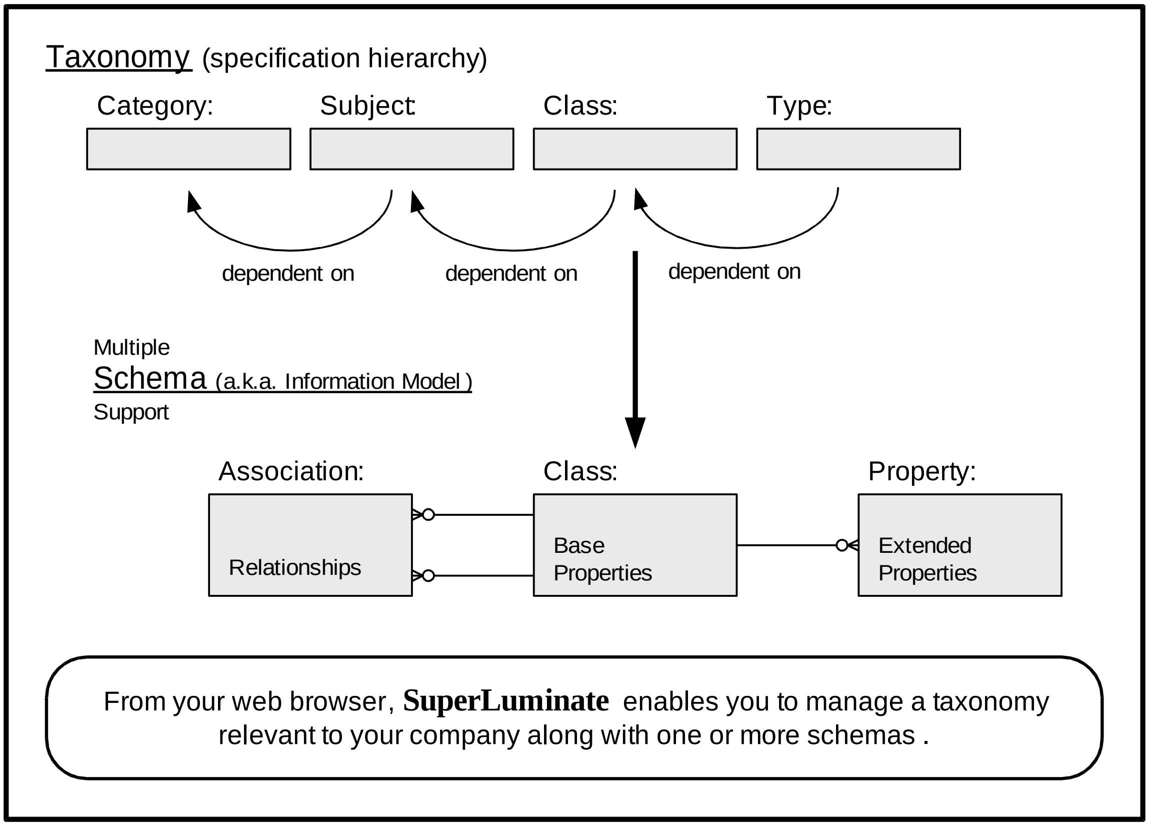
The definition (structure) of the taxonomy and the schema(s) define what the user can enter and maintain in the SuperLuminate data dictionary.

On the previous diagram there are two schemas, one schema with two disassociated Class objects and one schema with four Class objects associated in a hierarchical (parent / child) relationship.

Note:

> In the taxonomy you will also find Type along with Category, Subject, and Class. Type gives the user the ability to sub-type a Class object, e.g. for a Class object named “Table” the administrator may setup sub-types like “Development and “Production” allowing for the segmentation of Classes.

Taxonomy to Schema Interaction



From your web browser, **SuperLuminate** enables you to manage a taxonomy relevant to your company along with one or more schemas .

Data Abstraction

Data abstraction, as defined in the IRDS (Information Resource Directory System) standard.

Note: Level 0 information is in your everyday application system, e.g. customer name. Level 1, 2, and 3, information is entered and maintained in SuperLuminate.

Data Abstraction

Level	Architectural Model	Example	Owner
3	Metaschema (Physical)	Object	SuperLuminate
2	Schema (Information Model)	Column	Administrator
1	Metadata	Customer Name	User
0	Data	ACME	Organization

In reverse:

- Level 0 – defines the data in your everyday operational systems, e.g. accounting and customer tracking systems. The actual customer names and their account balances
- Level 1 – defines the definition of the data in Level 0
- Level 2 – defines the definition of the metadata in Level 1
- Level 3 – defines the definition of the schema in Level 2

Data Abstraction Level (0)

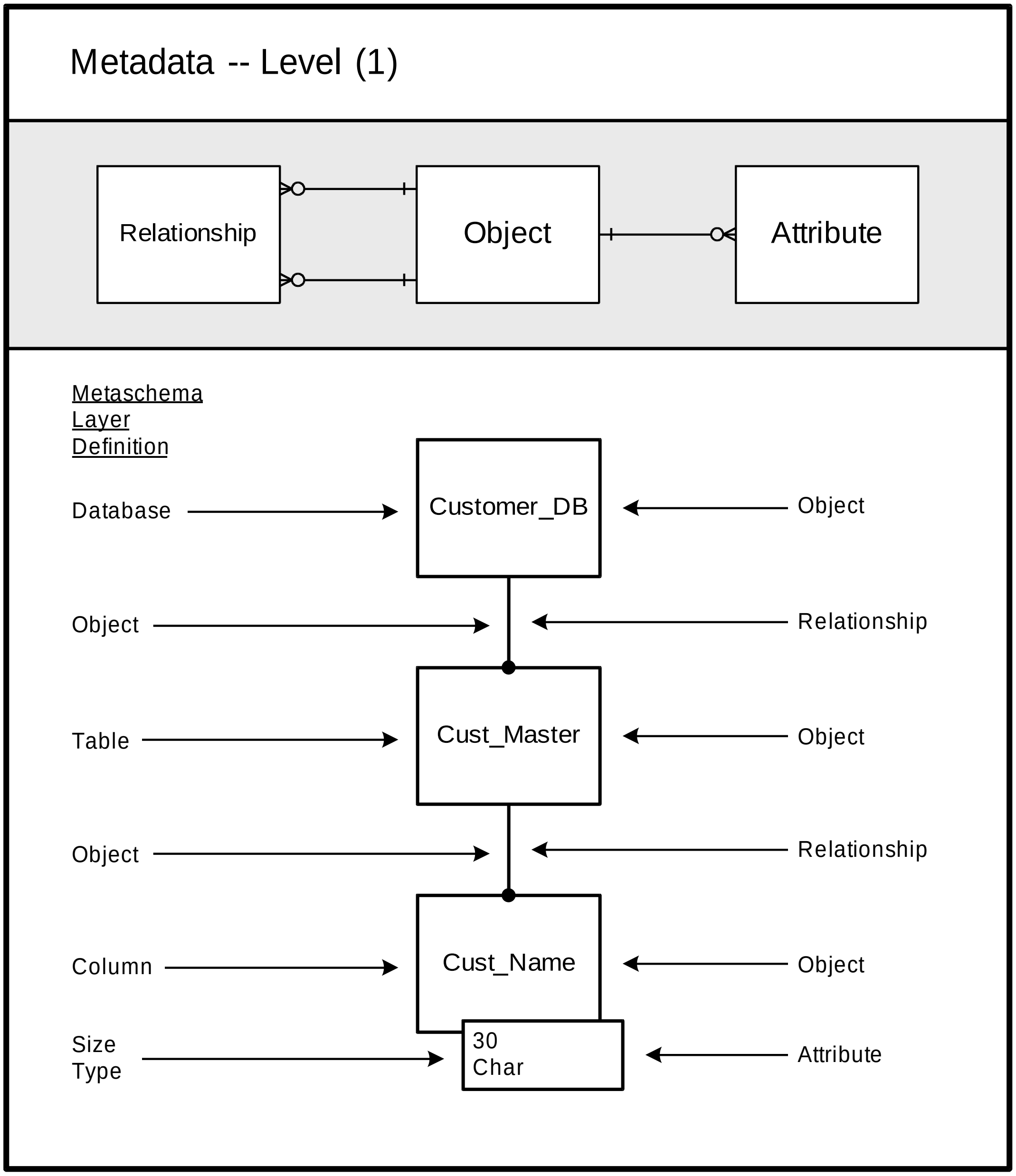
Level Zero: An organization's operational data. Not contained in SuperLuminate.

Data -- Level (0)

Level zero information is usually included and maintained in an application system like a sales systems. Examples of level zero data would be, a customers name like “John Doe”, a customers address like “1600 Pennsylvania Ave.”, a customers phone number like “(303) 555-1212”. This type of information can in some cases be added and maintain in the SuperLuminate data dictionary, but keep in mind that SuperLuminate was design to maintain data definitions, not application data values.

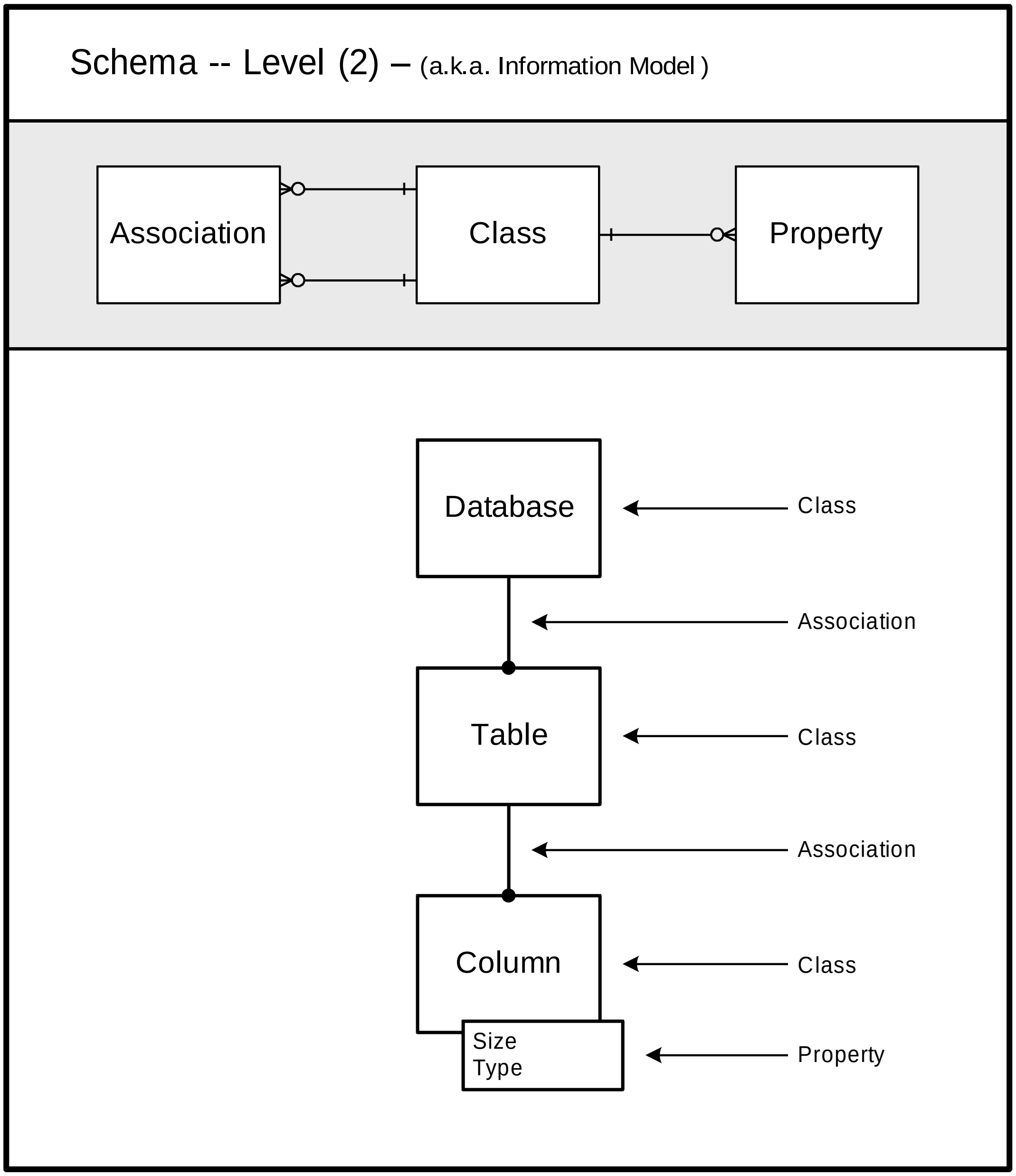
Data Abstraction Level (1)

Level One: The information that a SuperLuminate user enters and maintains in SuperLuminate, e.g. a glossary term.



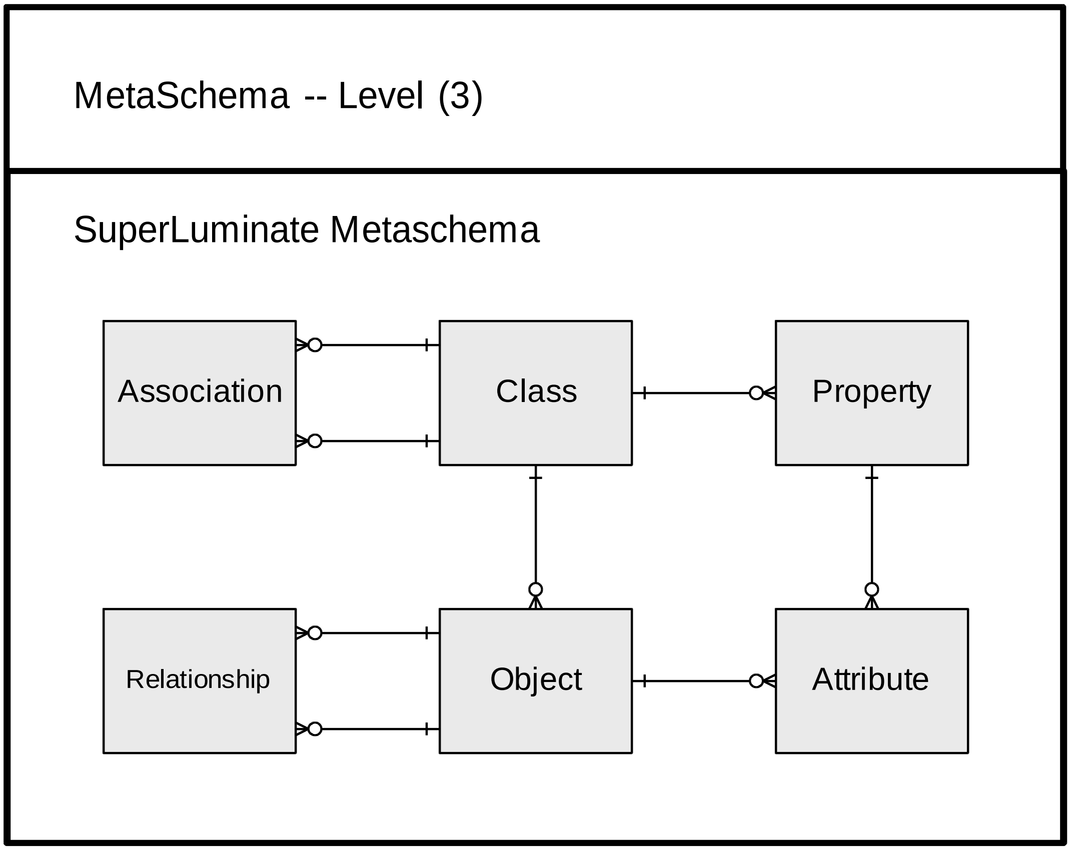
Data Abstraction Level (2)

Level Two: The information (control data) that your SuperLuminate administrator enters and maintains in SuperLuminate. This "control data" defines the taxonomy, schema (A.K.A. Information Model), SuperLuminate configuration components.



Data Abstraction Level (3)

Level Three: This is the internal physical definition of SuperLuminate.



Technical Architecture

Database Naming Conventions

Upper-case vs. Lower-case vs. Title-case

When installing SuperLuminate on a server that uses the Windows operating system, the case, e.g. uppercase or lowercase, of the table and column names is undistinguishable, but in a Linux/UNIX installation, SL_Class and SL_CLASS would be considered different tables. For this reason, stay with the convention set in the SuperLuminate install script.

Table Naming Convention

All table names in the SuperLuminate database start with an uppercase SL (for SuperLuminate) and a title-case name, e.g. SL_Class. It is important to keep the names in the case that they are provided to you out-of-the-box.

Column Naming Convention

All column names in the SuperLuminate database are in title-case, e.g. Created_DS, ended with a two character abbreviated class word.

Class Word Definition

All columns in the SuperLuminate database end with a class word code, e.g. DS (data and time stamp). Class words are used as a descriptive method to define the data format for a given column.

Note: The physical format (data type) will vary depending on the database system used, e.g. MySQL, Oracle, DB2.

Class Word	Format	Description
DT	date	Date only
TM	date	Time only
DS	date	Date and Time Stamp
DC	decimal	Real number
HX	varchar	Hex Characters
IT	integer	Integer
TX	text	varchar or long text
KY	key	Integer -- Generated key
FK	foreign key	Integer -- The key (KY) from another table
ID	identifier	Can be varchar or integer
FL	flag	char(1) Y or N
CD	code	char(10)

Versioning

Ontology (the nature of…) of Versioning

- Any registered user can create a new version of an existing object. A version has the same keys as the existing object with the version number incremented by one (1). The user can then update the non key properties and save the record.

- Hierarchical versioning (a parent object and all subsequent children) our copied and maintained as a version set

- By version group (an object assigned to a group) is when a version is assigned across a group of related objects, e.g. a group of objects scheduled for promotion or release all objects of this group would have the same version number.

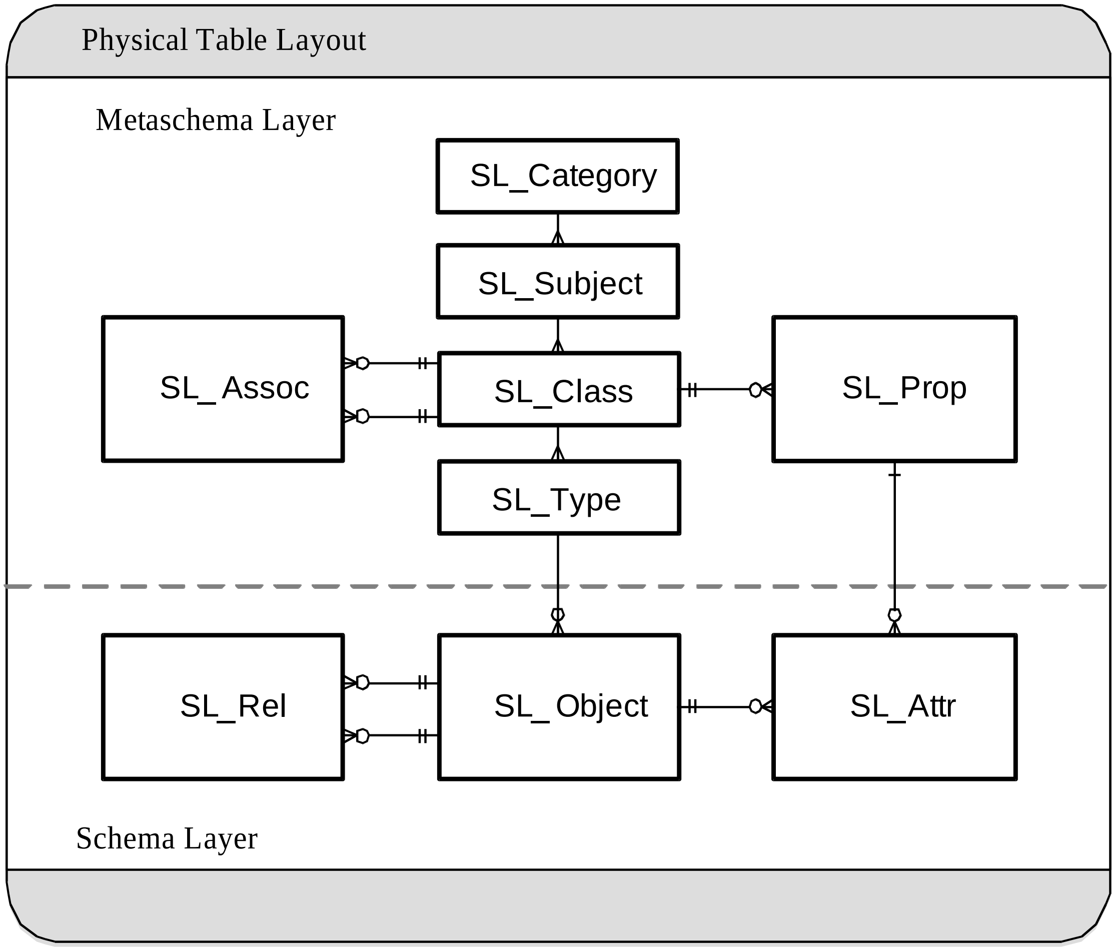
- By individual object. This is how versioning is currently implemented in SuperLuminate.

Copying

- Any registered user can create a new copy of an existing object. A copy does not have the same key as the copied record. The user can then update the any properties in the record and then save the record.

- The {Copy} function will copy all Extended Properties from the object but not the relationships to other objects.

Logical Model (not including auxiliary tables)



Note:

In the previous diagram there are two layers. In this guide the term metaschema is many times referred to as just schema, but in fact the metaschema layer contains the tables where the definition for the taxonomy and schema(s) that control user data entry are stored and maintained. The schema layer contains the tables where the user data (metadata) is stored and maintained.

The metaschema layer metadata is maintained by the SuperLuminate administrator and the schema layer metadata is maintained by the user.

Physical Table Definitions

Metaschema Tables

```
CREATE TABLE SL_Category (
      Category_ID             int(11) primary key auto_increment,
      Category_TX             varchar(30),
      Creator_ID              varchar(30),
      Created_DS              datetime,
      Changer_ID              varchar(30),
      Changed_DS              datetime,
      Status_CD               varchar(10));

CREATE UNIQUE INDEX SL_Category_A1 ON SL_Category (
      Category_TX);

CREATE TABLE SL_Subject (
      Category_ID             int(11),
      Category_TX             varchar(30),
      Subject_ID              int(11) primary key auto_increment,
      Subject_TX              varchar(30),
      Creator_ID              varchar(30),
      Created_DS              datetime,
      Changer_ID              varchar(30),
      Changed_DS              datetime,
      Status_CD               varchar(10));

CREATE UNIQUE INDEX SL_Subject_A1 ON SL_Subject (
      Category_TX,
      Subject_TX);

CREATE TABLE SL_Class (
      Category_ID             int(11),
      Category_TX             varchar(30),
      Subject_ID              int(11),
      Subject_TX              varchar(30),
      Class_ID                int(11) primary key auto_increment,
      Class_TX                varchar(30),
      Creator_ID              varchar(30),
      Created_DS              datetime,
      Changer_ID              varchar(30),
      Changed_DS              datetime,
      Status_CD               varchar(10));

CREATE UNIQUE INDEX SL_Class_A1 ON SL_Class (
      Category_TX,
      Subject_TX,
      Class_TX);

CREATE TABLE SL_Type (
      Category_ID             int(11),
      Category_TX             varchar(30),
      Category_DC             decimal(10,0),    -- drop down sort order
      Subject_ID              int(11),
      Subject_TX              varchar(30),
      Subject_DC              decimal(10,0),    -- drop down sort order
      Class_ID                int(11),
      Class_TX                varchar(30),
      Class_DC                decimal(10,0),    -- drop down sort order
      Type_ID                 int(11) primary key auto_increment,
      Type_TX                 varchar(30),
      Type_DC                 decimal(10,0),    -- drop down sort order
      Unique_FL               char(1),          -- for Future Use
      Creator_ID              varchar(30),
      Created_DS              datetime,
      Changer_ID              varchar(30),
      Changed_DS              datetime,
      Status_CD               varchar(10));

CREATE UNIQUE INDEX SL_Type_A1 ON SL_Type (
      Category_TX,
      Subject_TX,
      Class_TX,
      Type_TX);

CREATE TABLE SL_Prop (
      ROWID                   int(11) primary key auto_increment,
      Category_TX             varchar(30),
      Subject_TX              varchar(30),
      Class_TX                varchar(30),
      Property_TX             varchar(30),
      Datatype_CD             varchar(30),      -- for Future Use
      Length_IN               int(3),           -- for Future Use
      Default_TX              varchar(100),
      Rule_TX                 varchar(255),     -- for Future Use
      Creator_ID              varchar(30),
      Created_DS              datetime,
      Changer_ID              varchar(30),
      Changed_DS              datetime,
      Status_CD               varchar(10));

CREATE UNIQUE INDEX SL_Prop_A1 ON SL_Prop (
      Category_TX,
      Subject_TX,
      Class_TX,
      Property_TX);

CREATE TABLE SL_Assoc (
      ROWID                   int(11) primary key auto_increment,
      P_Category_TX           varchar(30),
      P_Subject_TX            varchar(30),
      P_Class_TX              varchar(30),
      C_Category_TX           varchar(30),
      C_Subject_TX            varchar(30),
      C_Class_TX              varchar(30),
      Assoc_Rule_TX           varchar(255),     -- for Future Use
      Creator_ID              varchar(30),
      Created_DS              datetime,
      Changer_ID              varchar(30),
      Changed_DS              datetime,
      Status_CD               varchar(10));

CREATE UNIQUE INDEX SL_Assoc_A1 ON SL_Assoc (
      P_Category_TX,
      P_Subject_TX,
      P_Class_TX,
      C_Category_TX,
      C_Subject_TX,
      C_Class_TX);
```

Schema Tables

```
CREATE TABLE SL_Object (
      ROWID                   int(11) primary key auto_increment,
      Category_TX             varchar(30),
      Subject_TX              varchar(30),
      Class_TX                varchar(30),
      Type_TX                 varchar(30),
      Name_TX                 varchar(100),
      Version_DC              decimal(7,3),
      Alias_TX                varchar(100),
      Desc_TX                 text,
      Usage_TX                text,
      Comment_TX              text,
      Group_CD                varchar(10),
      Source_ID               varchar(100),
      Creator_ID              varchar(30),
      Created_DS              datetime,
      Changer_ID              varchar(30),
      Changed_DS              datetime,
      Status_CD               varchar(10));

CREATE UNIQUE INDEX SL_Object_A1 ON SL_Object (
      Category_TX,
      Subject_TX,
      Class_TX,
      Type_TX,
      Name_TX,
      Version_DC);

CREATE TABLE SL_Rel (
      P_SL_Object_FK          int(11),
      P_Category_TX           varchar(30),
      P_Subject_TX            varchar(30),
      P_Class_TX              varchar(30),
      P_Type_TX               varchar(30),
      P_Name_TX               varchar(100),
      P_Version_DC            decimal(7,3),
      C_SL_Object_FK          int(11),
      C_Category_TX           varchar(30),
      C_Subject_TX            varchar(30),
      C_Class_TX              varchar(30),
      C_Type_TX               varchar(30),
      C_Name_TX               varchar(100),
      C_Version_DC            decimal(7,3),
      Creator_ID              varchar(30),
      Created_DS              datetime,
      Changer_ID              varchar(30),
      Changed_DS               datetime,
      Status_CD                varchar(10));

CREATE UNIQUE INDEX SL_Rel_A1 ON SL_Rel (
      P_Category_TX,
      P_Subject_TX,
      P_Class_TX,
      P_Type_TX,
      P_Name_TX,
      P_Version_DC,
      C_Category_TX,
      C_Subject_TX,
      C_Class_TX,
      C_Type_TX,
      C_Name_TX,
      C_Version_DC);

CREATE TABLE SL_Attr (
      SL_Object_FK             int(11),
      Category_TX              varchar(30),
      Subject_TX               varchar(30),
      Class_TX                 varchar(30),
      Type_TX                  varchar(30),
      Name_TX                  varchar(100),
      Version_DC               decimal(7,3),
      Property_TX              varchar(30),
      Value_TX                 varchar(100));

CREATE UNIQUE INDEX SL_Attr_A1 ON SL_Attr (
      Category_TX,
      Subject_TX,
      Class_TX,
      Type_TX,
      Name_TX,
      Version_DC,
      Property_TX);
```

Auxiliary Tables

```
CREATE TABLE SL_Event (
      ROWID                    int(11) primary key auto_increment,
      User_ID                  varchar(30),
      Event_CD                 varchar(30),
      Remote_IP_Addr_TX        varchar(255),
      Remote_Agent             varchar(255),
      Event_TX                 varchar(255),
      Created_DS               datetime);

CREATE TABLE SL_User (
      User_ID                  varchar(30) primary key,
      User_Type_CD             varchar(10),
      Password_TX              varchar(80),
      First_Name_TX            varchar(30),
      Last_Name_TX             varchar(30),
      Title_TX                 varchar(30),
      Email_Addr_TX            varchar(80),
      Phone_TX                 varchar(30),
      Company_TX               varchar(80),
      Dept_TX                  varchar(80),
      URL_TX                   varchar(80),
      Release_CD               varchar(10),
      Newsletter_CD            varchar(10),
      Creator_ID               varchar(30),
      Created_DS               datetime,
      Changer_ID               varchar(30),
      Changed_DS               datetime,
      Status_CD                varchar(10));

CREATE TABLE SL_Config (
      Config_ID                varchar(30) primary key,
      Color_Bkg_HX             varchar(6),
      Color_Msg_HX             varchar(6),
      Logo_Primary_TX          varchar(30),
      Read_Only_FL             char(1),
      Max_Rows_IN              int(5),
      Attr_01_TX               text, -- stores login page message
      Attr_02_TX               text, -- (Y/N) Ad Hoc SQL access
      Attr_03_TX               text, -- stores filestore path name
      Attr_04_TX               text, -- (Y/N) SuperLuminate lite access
      Attr_05_TX               text, -- stores MySQL binaries path name
      Attr_06_TX               text, -- customer Oracle connect string
      Attr_07_TX               text, -- System time out in seconds
      Attr_08_TX               text, -- for future use
      Attr_09_TX               text, -- for future use
      Attr_10_TX               text, -- for future use
      Creator_ID               varchar(80),
      Created_DS               datetime,
      Changer_ID               varchar(30),
      Changed_DS               datetime,
      Status_CD                varchar(10));
```

Software Specifications

SuperLuminate Folders and Application Software

This folder (sl -- SuperLuminate) contains all of the directories and files that makeup the SuperLuminate data dictionary application.

Folder: filestore

This folder contains all SQL files needed to create and maintain the SuperLuminate data structures, e.g. database and default control data.

- bulk_glossary_business_load.csv
  Sample of a business glossary that can be loaded via the import object functionality
- bulk_glossary_leisure_load.csv
  Sample of a leisure (boating) glossary that can be loaded via the import object functionality
- bulk_report_load.csv
  Sample reports that can be loaded via the import object functionality – these are reports that are loaded as metadata – the reports can be copied into and run via the reporting window
- install_schema_v113.sql
  Database schema that is used to create the underlying SuperLuminate data structure
- metadata_sl_attr_load.csv
  Optional metadata that defines the SuperLuminate system (self defining metadata) used by the [Flat File Adapters] SuperLuminate Metadata function – good example how full hierarchies of data is imported
- metadata_sl_object_load.csv
  Optional metadata that defines the SuperLuminate system (self defining metadata) used by the [Flat File Adapters] SuperLuminate Metadata function – good example how full hierarchies of data is imported
- metadata_sl_rel_load.csv
  Optional metadata that defines the SuperLuminate system (self defining metadata) used by the [Flat File Adapters] SuperLuminate Metadata function – good example how full hierarchies of data is imported
- sl_db_back_20100427.sbk (example – not distributed)
  This file is actually not distributed with SuperLuminate but is an example of a database export (backup) that can be generated via the [Flat File Adapters] page

Folder: image

This folder contains all of the images used by SuperLuminate including sample company logos.

Folder: jsrs

This folder contains the Java programs used to control the cascading menu selections on pages like [Add] page gotten to via the (New Entry) button.

Folder: session

This folder contains all user session files. Only active sessions should be in this folder. Any sessions that are older than 24 hours can be deleted.

File: index.html

This HTML program is used to launch the SuperLuminate application if no directory parameter is specified.

File: sl_about.html

This HTML file contains a description of the SuperLuminate application. It is the "About Box" and can be accessed from the top of any page within the SuperLuminate application via the "About" hyperlink.

File: sl_colorchart.html

This HTML file contains a set of colors and hex value equivalents that can be used to set background and message colors. This is only a sample set, it does not contain anywhere near all of the color values that can be used to set background and message line colors.

File: sl_global.php

This file contains all of the global variables used to connect the SuperLuminate database along with other variables used to control the functions in SuperLuminate.

File: sl_help.html

This HTML file contains the basic help (getting started) information for the SuperLuminate application. It is the "Help Box" and can be accessed from the top of any page within the SuperLuminate application via the "Help" hyperlink.

File: sl_license.html

This HTML file contains the license information for the SuperLuminate application. The license is a copy of the standard Open-Source GPL.

File: sl_main.php

This PHP program is the SuperLuminate application.

File: sl_report.php

This PHP program is to generate the external report page when you hit the report button on one of the display pages.

Administration Topics

User Maintenance (adding or modifying a user)

- Add and modify users via the [Manage User] page
- [Main Menu] à [Administration] à [Manage User]

Schema Management

- Change the SuperLuminate schema [Manage Schema] page. The administrator will maintain the schema definitions in SuperLuminate. The schema specifications (taxonomies and metaschemas) only affect the [Add] page where a user will create new records. On the [Add] page the taxonomies control the values in the drop down picklists. The metaschemas control whether an object can be associated to another object as a parent or child and whether the object has extended properties.
- [Main Menu] à [Administration] à [Manage Schema]
    - Taxonomy
        - Add, Delete, and Modify Categories
        - Add, Delete, and Modify Subjects
        - Add, Delete, and Modify Classes
        - Add, Delete, and Modify Types
    - Schema(s)
        - Add, Delete, and Modify Associations
        - Add, Delete, and Modify Properties

System Configuration

- Change the look and feel of SuperLuminate via the [Modify Configuration] page
- [Main Menu] à [Administration] à [Modify Configuration]

Field to Adjust Look and Feel:

Configuration ID

- Is for future use – will allow several saved settings

Background Color

- The backgrounds color setting defines the color of each page. Use Hex Color Chart for ideas.

Message Color

- The message color setting defines the color of information message line at the top each page. Use Hex Color Chart for ideas.

Primary Logo

- The primary logo define the logo to be displayed at the top of each page. Any logo can be selected as long as the logo exists in the sl/image directory (folder).

Default Values:

Background Color: E0FFFF
Message Color: FFFF66
Primary Logo: superluminate_logo.gif

Look and Feel Examples:

Background Color: FFDEAD
Message Color: E0FFFF
Primary Logo: stanford_univ_gsb_logo.gif

Background Color: FFDEAD
Message Color: E0FFFF
Primary Logo: purdue_univ_logo.gif

Background Color: FFDEAD
Message Color: E0FFFF
Primary Logo: hitachi_logo.gif

Background Color: F0F8FF
Message Color: 99CCFF
Primary Logo: washmutual_logo.gif

Other Configuration Fields:

Read Only Enabled:

- Y

    - Displays the read only button on the login page so users without a user id and password can use the system
- N
    - Remove the read only button from the login page

SL Lite URL Enabled:

- Y
    - Enables the use of the SuperLuminate Lite Glossary page. This page is access via the same URL are the current configuration of SuperLuminate with the following appended (?Glossary)
    - Example: http://localhost/sl/sl_main.php?Glossary
- N
    - Disables the use of the SuperLuminate Lite Glossary.

Ad Hoc SQL Enabled:

- Y
    - Enables a registered user (has user id and password) to create and execute ad hoc SQL select queries via the [Reporting] page. Note: SELECT queries only.
- N
    - Disables the ability to create ad hoc SQL queries. Note: The admin user can create and execute any form of SQL query via the AD HOC SQL Window. This field does not disable admin access.

Max Rows Returned:

- Integer Number, e.g. 2000
    - Determines the maximum number of rows that can be retrieved from the SuperLuminate database in one report. This number can be increased or decreased depending on system performance and browser memory.

Login Memo:

- Any text message
    - This is the message you want displayed in the message bar on the login page.

External File Path:

- Valid path name

    - This is the path name to be added to any statement that needs access to a file in the sl/filestore directory (folder). Used for loading and exporting data.

Database Binaries Path:

- Valid path name
    - This is the path name to be added to any statement that needs access to directory (folder) containing the database binaries. Used for backup and recovery of the SuperLuminate database.

Oracle Connection String:

- Valid Oracle Connection String With Embedded Variables
    - Example
        - (DESCRIPTION= (ADDRESS= (PROTOCOL= TCP) (HOST= $host) (PORT= $port)) (CONNECT_DATA= (SID= $sid)))Example
        - Note the three variables that need to be in place
    - This allows you to customize the connection string for your company if it is required – SuperLuminate defaults the connection string to the above example – it has worked for all but a few Oracle databases.

System Time Out in Seconds:

- Integer
    - PHP defaults to 30 – SuperLuminate defaults to 50
    - Zero (0) set the system time out to infinite
    - If when running a database import, backup, or any other SuperLuminate function or report and you receive a system timeout message boost this number.
    - Importing large Oracle schemas may require you to set this time as high as 500 seconds.

Database Backup and Recovery

- Backup and restore the SuperLuminate database via the [Backup Restore Adapters] page

- [Main Menu] à [Administration] à [Backup Restore Adapters]

System Monitoring

- Monitor the SuperLuminate system via various reports on the [Management Reporting] page

- [Main Menu] à [Administration] à [Management Reporting]

External Metadata Importing and Exporting

- Load external metadata into the SuperLuminate database via the [Flat File Adapters] page

- [Main Menu] à [Administration] à [Flat File Adapters]

NOTE:

When loading metadata into SuperLuminate count the number of rows in the import file. Once the import is complete via the [Reporting] page select the (Schema Count) button to make sure all of your records were loaded correctly. If no load error occurred and the count is lower than you expected there were probably duplicates (duplicate objects keys) in your import file. Correct the import file and rerun the import. All previously imported records will be updated using the object key to match existing records.

Make sure that double quotes are not within data fields. Use singe quotes in data fields.

If the last record in the input file does not load make sure that the last record is ended with a carriage return. A carriage return indicates the end of record to be loaded.

Load File Formats

Import Object **-- bulk_glossary_business_load.csv**
-- bulk_glossary_leisure_load.csv
-- bulk_report_load.csv

```
REPLACE INTO TABLE SL_Object
Category_TX, Subject_TX, Class_TX, Type_TX, Name_TX,
Version_DC, Alias_TX, Desc_TX, Usage_TX, Comment_TX
```

Export Object **-- bulk_glossary_export.csv**

```
SELECT
        Category_TX,
        Subject_TX,
        Class_TX,
        Type_TX,
        Name_TX,
        Version_DC,
        Alias_TX,
        Desc_TX,
        Usage_TX,
        Comment_TX)
INTO OUTFILE '$filename'
FIELDS TERMINATED BY ','
OPTIONALLY ENCLOSED BY '\"'
LINES TERMINATED BY '\r\n'
FROM SL_Object
WHERE Class_TX = 'Glossary'
ORDER BY
Category_TX,
Subject_TX,
Class_TX,
Type_TX,
Name_TX,
Version_DC";
```

SuperLuminate Metadata

After all of the following three files are loaded a set of SQL code embedded in the SuperLuminate application will create internal keys joining all the data from table to table. This Join creates the parent child links between objects and added the extended properties.

Load metadata_sl_object_load.csv

```
REPLACE INTO TABLE SL_Object
Category_TX, Subject_TX, Class_TX, Type_TX, Name_TX,
Version_DC, Alias_TX, Desc_TX, Usage_TX, Comment_TX
```

Load metadata_sl_rel_load.csv

```
REPLACE INTO TABLE SL_Rel
P_SL_Object_FK, P_Category_TX, P_Subject_TX, P_Class_TX,
P_Type_TX, P_Name_TX, P_Version_DC, C_SL_Object_FK,
C_Category_TX, C_Subject_TX, C_Class_TX, C_Type_TX,
C_Name_TX, C_Version_DC
```

Load metadata_sl_attr_load.csv

```
REPLACE INTO TABLE SL_Attr
SL_Object_FK, Category_TX, Subject_TX, Class_TX, Type_TX,
```

Name_TX, Version_DC, Property_TX, Value_TX

End of User Guide

For more information go to WWW.SuperLuminate.com or email query@superluminate.com

-- This page was intentionally left blank