

# **InfoLibrarian™**

## **Metadata Stewardship User Guide**

## **Technical Metadata**

Technical Metadata supports developers and technical personnel in linking tools, applications, and systems into a solution. For example, it explains database structure, installed applications, and server systems..

## **Technical metadata provides details about**

Source destination information

Data element definition

Impact analysis

Lineage analysis

Data dictionaries

Audit ability- by capturing entitlement, definitions and occurrences of data points within your systems

## **Business Metadata**

InfoLibrarian™ presents Business Metadata in the following ways. Graphical representation of subject matter and categories that your users can drill through.

Business views present descriptions and names in the terminology understood by the business users. Visual maps that show the inter-relationships between the different data points within IT systems and you can categorize it the way you want.

**Business Metadata** describes technical computer data and applications in a way that is understandable to the business users. Business Metadata hides technological constraints by mapping business language to the technical systems.

Context

Understandability

Search ability

Usability

## **Business Metadata answers the following questions:**

What are the valid values?

Where did this data come from?

How do I get more information?

Is there data that can help me?

## **Business Metadata provides users with logical drilldown:**

Are there any reports for me?

Is this table going to work?

Show an inventory of reports.

How good is the data?

Is this data suitable for my analysis?

Is the data fresh?

How current or when was it last updated?

Is there a single version of the truth?

## **What is A Business Term?**

A business term is any term that is familiar to the business as opposed to technical terms which are used by technical people. However, both may describe the same thing in different language. Simply stated, often the main hindrance to business people using computer based systems is that they were built by technical people and the terminology tends to be technical making them difficult to understand.

**InfoLibrarian Business Term** is analogous to a common comprehensible term that can be applied to an artifact in metadata. In other words a Business Term is a "Common" expression known to the business for a thing. Describes the object types and tool feature that allows you to define and manage metadata semantics and to materialize business views of the technical metadata within the portal.

## **With the InfoLibrarian Business Terms you can Define and map :**

- Hierarchies.
- Relationships.
- Many things that have the same meaning.
- Things that have many meanings.
- Semantic definitions for artifacts.
- Semantic classifications for artifacts.
- Semantically controlled vocabularies for artifacts.

### **Note:**

*A Business Term is the starting point for building semantics about a particular piece of data. Terms are then mapped to other physical entities, categories and subject areas, even documents and web links.*

*Business Term object types are user definable and you may create your own special term types.*

## **Business Rule**

A business rule defines or constrains one aspect of your business that is intended to assert business structure or influence the behavior of your business.

Some business rules focus on the policies of your organization

## **What is a Business Rule?**

A business rule is guidance that there is an obligation concerning conduct, action, practice, or procedure within a particular activity or sphere.

## **Business Rule Types**

Domain membership asserts that an attribute's value is always taken from a previously defined data domain. For example, an online catalog vendor may specify a domain of fabric colors, then assert that all sweaters that can be ordered online must be of one of the named colors.

A mapping membership rule asserts that the relation between two attributes or fields is restricted based on a named mapping. An example enforces the mapping from U.S. State name to its corresponding two-letter postal abbreviation.

A completeness rule specifies that when a condition is true, a record is incomplete unless all attributes on a provided list aren't null. An example in the financial world would specify that if the security being traded is a stock option, the trade is incomplete unless a strike price and expiration date are provided.

An exemption rule says that if a condition is true, then those attributes in a named list shouldn't have values. For example, if the customer's age is less than 16, then the driver's license field should be null.

Consistency refers to maintaining a relationship between two (or more) attributes based on the content of the attributes. A consistency rule indicates that if a particular condition holds true, then a following consequent must also be true. An example in a credit analysis application might say that the amount allowed for a monthly mortgage payment must be no more than 35 percent of the monthly gross income.

## **Taxonomy**

The term taxonomy has been widely used and abused to the point that when something is referred to as taxonomy it can be just about anything, though usually it will mean some sort of abstract structure.

The benefit of this approach is that it allows related terms to be grouped together and categorized in ways that make it easier to find the correct term to use whether for searching or to describe an object.

Note that the taxonomy helps users by describing the subjects; from the point of view of metadata there is really no difference between a simple controlled vocabulary and taxonomy. The metadata only relates objects to subjects, whereas here we have arranged the subjects in a hierarchy. So taxonomy describes the subjects being used for classification, but is not itself metadata.

*Within InfoLibrarian, Taxonomies are built using Subject Areas and Categories.*

## **Controlled Vocabulary**

Controlled vocabulary is a rather broad term, but here we mean by it a closed list of named subjects, which can be used for classification. In library science this is sometimes known as an indexing language. The constituents of a controlled vocabulary are usually known as terms, where a term is a particular name for a particular concept. (This is pretty much the same as the common-sense notion of a keyword.)

The purpose of controlling vocabulary is to avoid authors defining meaningless terms, terms, which are too broad, or terms, which are too narrow, and to prevent different authors from misspelling and choosing slightly different forms of the same term.

*Within InfoLibrarian controlled vocabularies are built using Business Terms.*

## Brief Explanation of Objects, Properties and Mappings

Within InfoLibrarian each piece of metadata is stored as an object. In fact the repository is object oriented by design. When metadata is created and stored within InfoLibrarian, an object type is created for the piece of metadata being stored. For example: a database columns' metadata is stored using an object type of type "DBMS Column". The object type "DBMS Column" may contain various properties such as precision, scale and data type. Think of the object type as a container or template for the metadata. Object types can be hierarchical as well. Given the previous example the "DBMS Column" object type can be a child of the "DBMS Table" object type.

When we store a piece of metadata using an object type we are actually storing an instance of a piece of metadata at a particular point in time. This is important because, the power of the object oriented paradigm will allow us to represent the real world without constraints.

Basically, the way you implement and arrange your object types becomes your meta-model.

Scanners create all the object types and instances metadata for you and you can define your own object types at any time using the Object Modeler or InfoLibrarian Administrator.

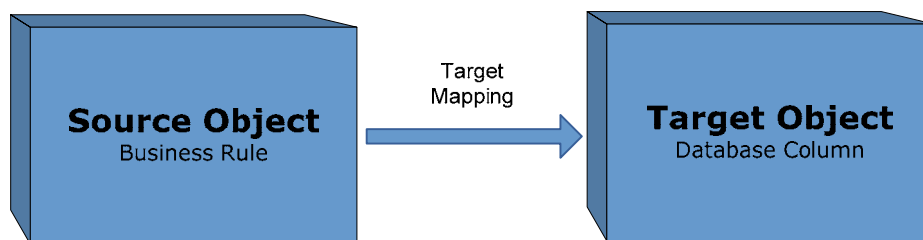
## Mappings

Once an object instance is stored (by populating an object type), we can then map that object to any other object within the repository. There are two kinds of mappings: Source and Target.

The mapper tool can be launched from the properties page of any object within the Administrator. This tool allows you to select from the repository metadata and map it to the selected object either as a source or target mapping.

Mappings direction is really not important from a functionality perspective because the portal is designed to show these relationships in either direction seamlessly. For example if something maps to me it will infer it even though I am not explicitly mapped to the something. You do not need to create bidirectional mappings. It is best to pick a strategy and keep it simple and consistent.

## Inferred Mappings



In this scenario, when viewing the target object. The Source object (Business Rule) source mapping will be indicated as inferred within the tool.

## Target Mappings

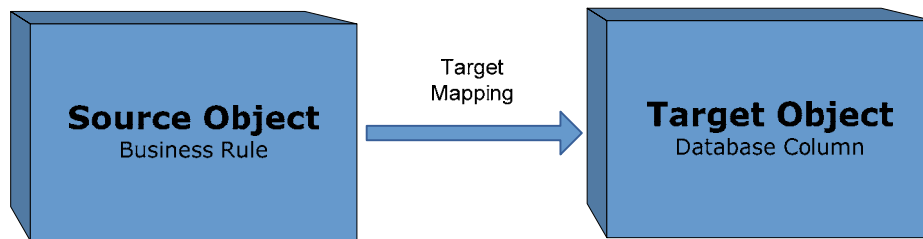
In many cases you need only use the target mappings to create your mappings. The following are examples of where target mappings are most used:

Business Terms → DBMS Columns

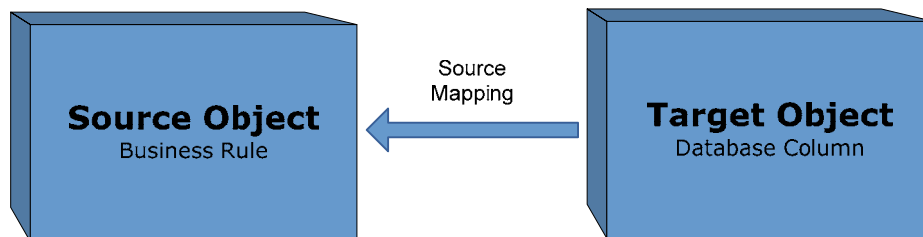
Subject Area Category → Business Terms

Business Rule → Metadata Object

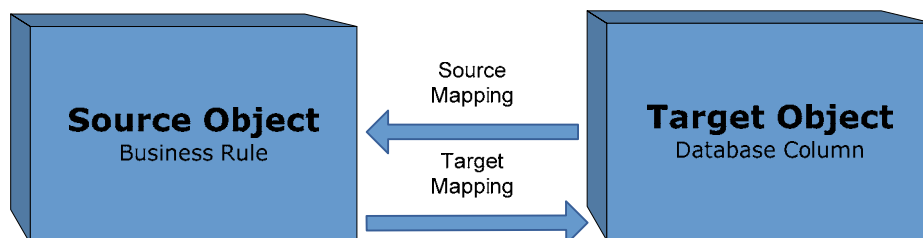
**Scenario One** – This is all that is required



**Scenario Two** – Depends how you want to look at it



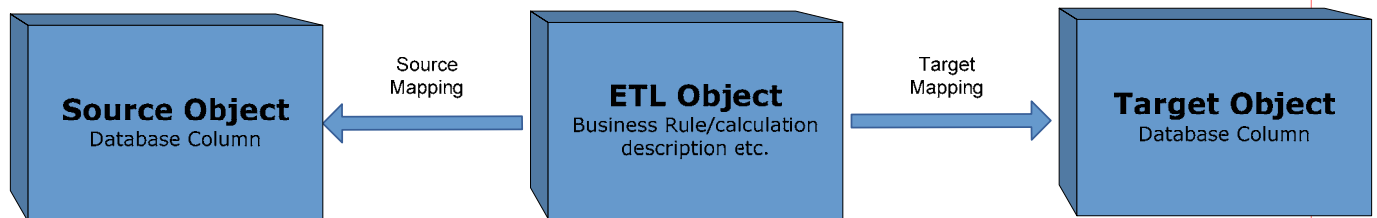
**Scenario Three** – This would be very verbose



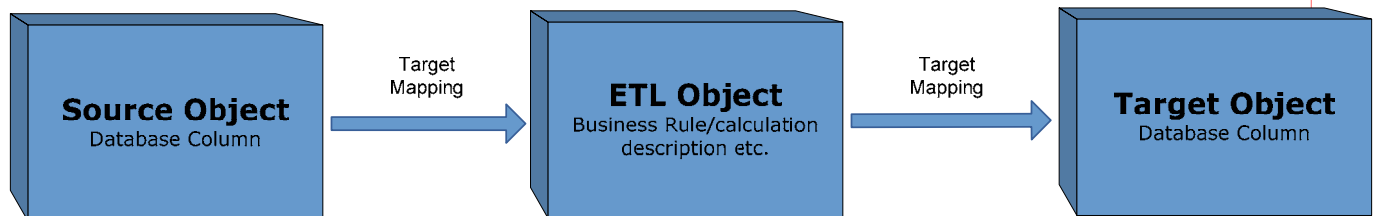
## Source Mappings Defining ETL Processes

Typically source mappings would be used in a case where you were trying to define lineage processes such as ETL. The way this is done is to create an ETL Object type that can represent the metadata for your process, then map the ETL Object to its source and target (columns possibly) by using both source and target mappings respectively. In this way the transformation logic or calculation can be represented as an intermediate object which resembles how the ETL exists in the real world.

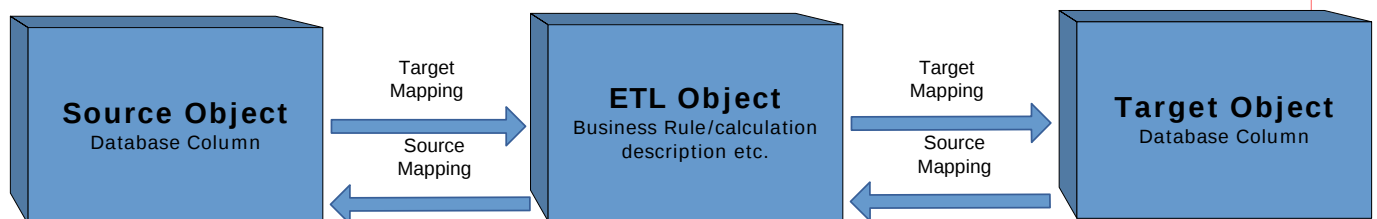
### Scenario One – Create ETL Metadata Source and Target Mappings



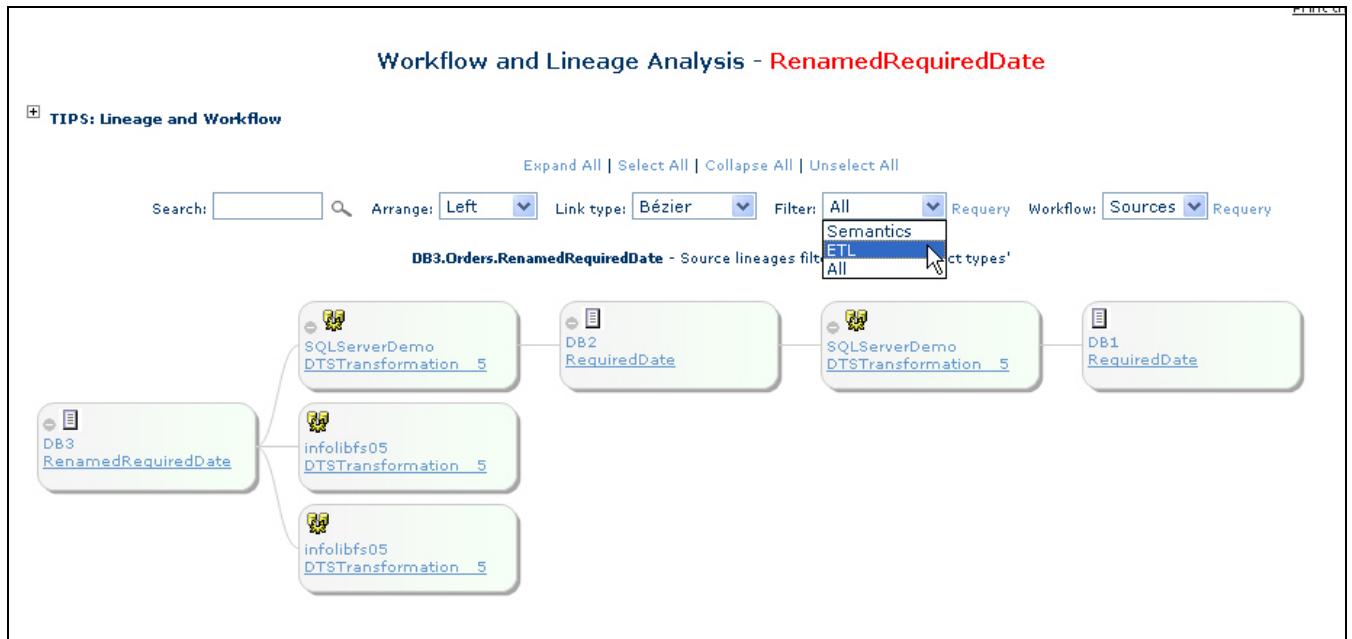
### Scenario Two – Create ETL Using Target Mappings Only



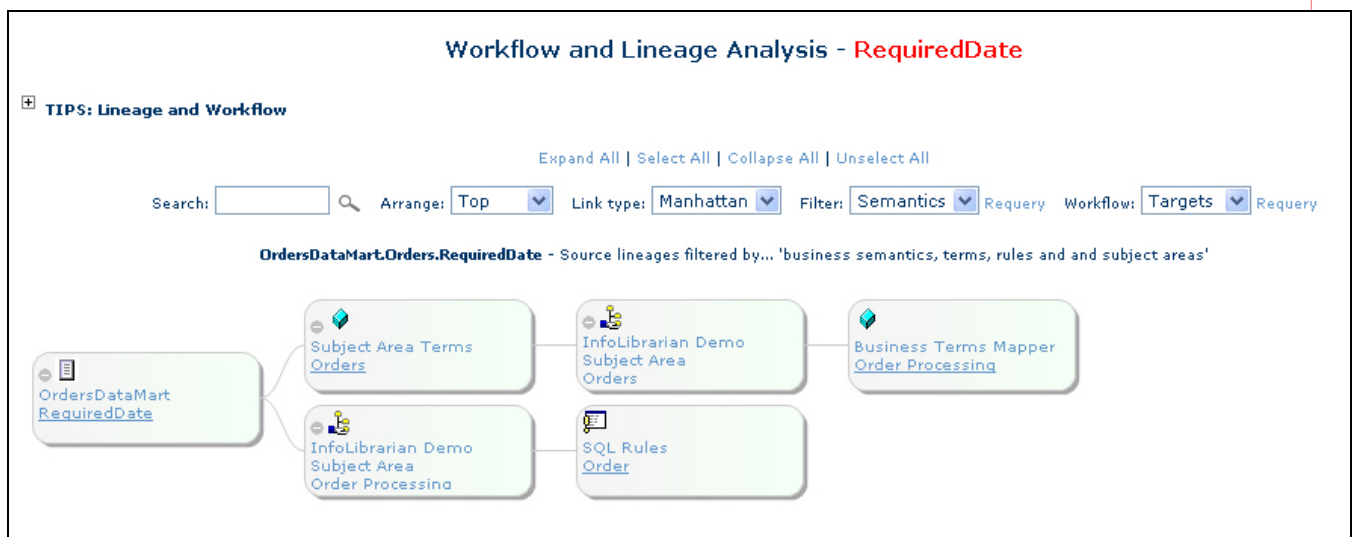
### Scenario Three – Create ETL Using both Mappings



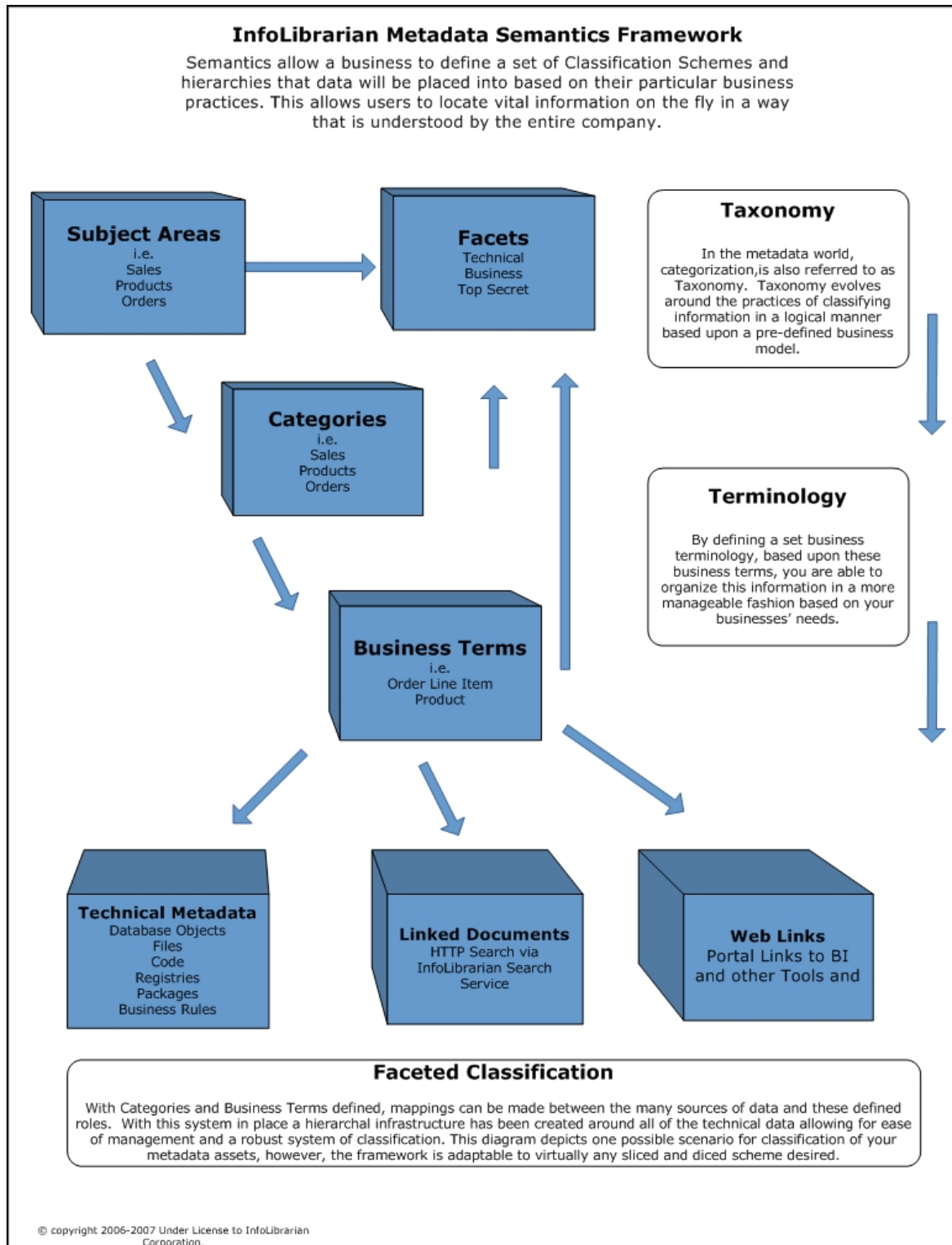
**Example of the Lineage Display in the portal** - this is calculated by InfoLibrarian from the mappings in the repository.



**Example of the Lineage Display in the portal** – This example shows only semantics related mappings and workflow.



## The Components of the InfoLibrarian Semantic Framework



## Metadata Semantics – Use Case

The screenshot shows the InfoLibrarian Administrator (4.1 Desktop Data Steward - INFOLIBRARIAN\BBREWER) interface. The left pane displays a tree view of the metadata hierarchy, with 'COUNTRY\_ID' selected under 'DBMS Column'. The right pane shows the details for the 'COUNTRY\_ID' object, including its name, type (DBMS Column), and last changed information. Below this, there are fields for 'Keyword Description' and 'Comments', and a 'Save Changes' button. At the bottom, a table titled 'Scanned Properties (Read Only)' lists various properties and their values.

| Property Name      | Property Value | Last Changed in Repository |
|--------------------|----------------|----------------------------|
| Data Type          | CHAR           | 11/28/2005 7:02:57 PM      |
| Data Length        | 2              | 11/28/2005 7:02:57 PM      |
| Character Set Name | CHAR_CS        | 11/28/2005 7:02:57 PM      |
| Column Position    | 1              | 11/28/2005 7:02:57 PM      |

The diagram above depicts an example of metadata that has been loaded as the result of an Oracle scan. At this level, we would consider this to be technical metadata. The country\_id column has these particular properties and physical column description. Notice that the specific type of metadata is related specifically to the physical implementation of this particular database column. Data Type etc are physical metadata attributes.

While this type of information is very useful, especially to the technical team, it may or may not represent any real meaning to a typical business user. Nor does it show any information about where it came from, how it got there or what it is used for. This is simply a data element description at this point.

Put simply... we need to present metadata to the business in non-technical terms. And to show exactly the who, what, when, where and how about this particular piece of data.

**Also consider the following scenario:**

Suppose this data element named Country\_Id exists in other locations and depending on where it is, it may or may not even have the same meaning or be used for the same purpose. It may not be the same piece of data or even be sourced from the same location. Further, the processes that move and manipulate this particular piece of data may be complex and involve transformations or calculations. Perhaps, it is viewed by different communities in different ways depending on how they use it

*Metadata Semantics and Business Metadata helps us to address these problems and allows us to represent real world complexities and points of view in a way that is useful*

**One Method of defining Business Metadata.**

Once scanning has been completed, it is possible to extend the metadata objects that have been loaded into InfoLibrarian. Custom properties can be defined for any object type within the repository.

Let's take the country\_Id column for example:

Theoretically, this column may belong to a particular system, be maintained by a particular user and have a particular business description.

By adding three new properties to the "DBMS Column" object type as follows:

System

Steward

Business Description

We can easily capture the business metadata for database columns in this example.

**Caveat**

Should semantics come into play in your environment, the best option for defining business metadata is to implement the semantics techniques mentioned in the next section before spending too much time performing data entry.

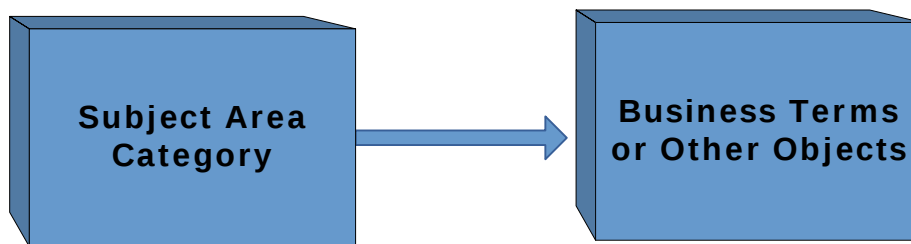
## Metadata Semantics

The following elements together form the working objects that will help you to define and capture business metadata and semantics within your environment.

- Business Terms
- Business Rules
- Subject Areas
- Categories
- Mappings

## The Semantic Process

Essentially, the idea is to either define or capture a list of terms (Not necessarily “Business Term” Object Types) and their mappings to other technical metadata (already scanned perhaps). At the same time can then assign the terms themselves to taxonomy. This is accomplished by further mapping the terms to categories and sub-categories you can define within a subject area.



**Note:** When any object is mapped to a Business Subject area Category, it is displayed by default as a “See Also hyperlink” to the target object.

## Subject Areas and Categories

**Business Subject Areas** are in fact a special type of metadata catalog. You may define as many subject areas as you need. Each subject area being the highest level of a set of categories. IE IT, Management, Division etc.

**Subject Area Categories/Sub-Categories** is in fact object types. Therefore a category is in fact an instance of an object type. Each level is a sub-object type of the parent object type.

*The built in Object Type Hierarchy structure for a Subject Area is as follows:*

- Business Subject Area Top Level
  - Business Subject Area Sub Level
    - Business Subject Area Sub Sub Level

## Subject Areas Continued...

You are free to define your own hierarchies using the object modeler or Administrator. Each level in the subject area (category, sub category etc...) is another object type. You can have virtually infinite arrangements of hierarchies or taxonomies made up of limitless object types. When you add a level to a subject area, you will need to select or define the object type of the level you are creating.

This is a very flexible and powerful functionality of the InfoLibrarian product. Because Categories are in fact objects, you can map them to other objects... like Business Terms, Linked Documents or any other objects you define.

Many classification schemes can be built using the semantics capabilities around the same metadata providing you with the capacity to define multiple perspectives of your metadata to different user communities.

In other words, users can see information from their point of view of the world. This is further enhanced with the Portal Perspectives which allow you to publish these perspectives to each community with their own web URL, look and feel.

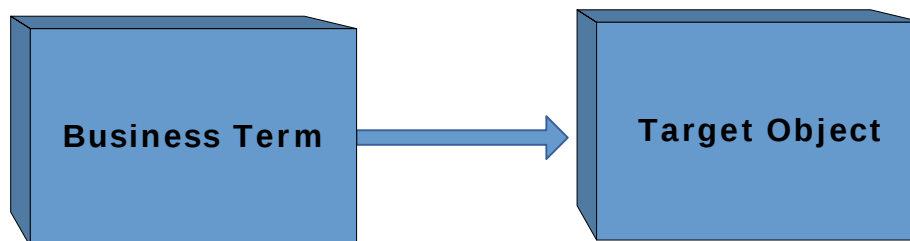
## Defining Your Business Terms

Business Terms when used are the pivotal element for establishing semantics.

By defining a business term, we are basically creating a well known name for a particular piece of data. I.e. "Order Processing Code" could be a particular business term. This term can then be mapped to several instances of this piece of data as follows:

DatabaseOne -> OrderId  
DatabaseTwo -> OrdCode  
DatabaseThree -> OrderProcCode

Notice that what we are doing here is simply linking the 3 instances of this data element to a common term.



Note: Business Terms can have additional metadata as well. For example, the description or any other custom property such as "Data Steward" or "Person Responsible" or "System" as further examples.

## More Semantics on Business Terms

Let's say that you are using a modeling driven architecture and the Logical data elements from your data model are loaded into InfoLibrarian from a scan. There would be no need to recreate an entire list of terms, since they were defined at design time and stored in your modeling tool.

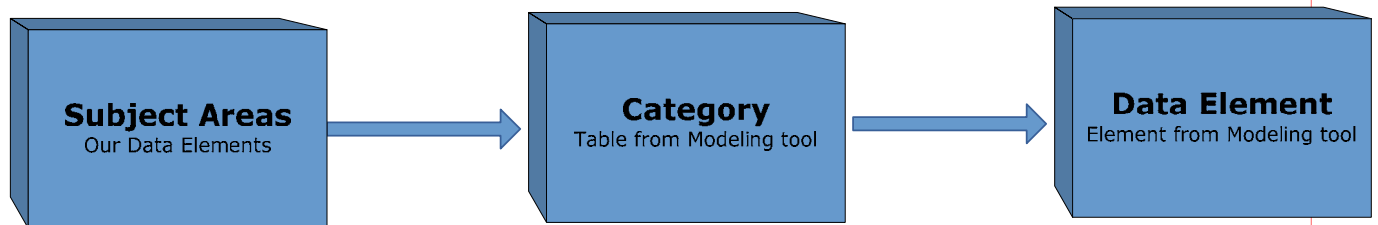
In this case we might be able to assume that the logical element name is in fact the business term (from your point of view). Since InfoLibrarian allows you to map anything to anything else, you don't necessarily have to use the actual pre-defined business term objects as business terms. You may use any object. In this example we'll use

"Modeling Element" as the business term. Since the modeling tool scanner loads logical data elements into the "Modeling Element" object type.

Using the above example we could then simply map the following directly to the instance of the "Model Element" already in the repository named "Order Processing Code".

```
DatabaseOne -> OrderId  
DatabaseTwo -> OrdCode  
DatabaseThree -> OrderProcCode
```

Then Create Categories based on the Modeling Entities (Table Names) in a Subject Area then map the Modeling entities to the Categories. (See later where you can actual automate this)



*The Data Element would map to its target physical columns.*

This shows that the flexibility is virtually unlimited. You will need to examine each scenario and determine your own subject areas, categories, semantic arrangements, object types, taxonomies, business terms etc... In any case, it is best to start with a finite prototype to materialize your results that you will become more familiar with how all of the semantics are materialized in the portal before taking on more complex structures.

### NOTE:

The pre-defined "Business Term" object type is intended to be used when no business terms exists or for convenience. However, you can use any object type as the "So called business term" if this makes sense in your scenario. Remember its all semantics.

You may need to create your own custom business term object types if you intend on defining unique custom properties for different categories of business terms. These user defined objects are created within user defined catalogs.

## Is It Possible to Automate Some of This?

There are several ways to minimize the manual effort involved with defining metadata semantics, mappings, terms and categories.

1. Use the Mapping wizard to rapidly map items by search criteria.
2. Create Spreadsheet templates to load the information containing the business term, mappings, category and subject area.
3. Customize the scanners themselves to automatically create categories and mappings on load.

### 1. Mapping Wizard

The Mapping Wizard allows you to search for potential mappings for an object based on your search criteria.

| Object (Parent.ObjectName)                                                     | Metadata Source          | Metadata Catalog          | Object Type           |
|--------------------------------------------------------------------------------|--------------------------|---------------------------|-----------------------|
| <input type="checkbox"/> DB1.Orders                                            | SQLServerDemo            | DB1                       | DBMS Table            |
| <input type="checkbox"/> DB2.Orders                                            | SQLServerDemo            | DB2                       | DBMS Table            |
| <input type="checkbox"/> DB3.Orders                                            | SQLServerDemo            | DB3                       | DBMS Table            |
| <input checked="" type="checkbox"/> ElementContent.Model.Namespace.contents... | XML Compliant Tools      | XML                       | Model/Attribute       |
| <input checked="" type="checkbox"/> Northwind.Orders                           | Northwind Traders Sam... | Northwind                 | DBMS Table            |
| <input checked="" type="checkbox"/> Northwind.Orders                           | SQLServerDemo            | Northwind                 | DBMS Table            |
| <input type="checkbox"/> OrdersDataMart.Orders                                 | SQLServerDemo            | OrdersDataMart            | DBMS Table            |
| <input type="checkbox"/> Organization.Orders                                   | Business Subject Areas   | InfoLibrarian Demo Sub... | Business Subject Area |
| <input type="checkbox"/> SQL Rules.Order                                       | Business Rules           | SQL Rules                 | Business Rule         |
| <input type="checkbox"/> Subject Area Terms.Order Number                       | Business Terms           | Subject Area Terms        | Business Term         |
| <input checked="" type="checkbox"/> Subject Area Terms.Orders                  | Business Terms           | Subject Area Terms        | Business Term         |
| <input type="checkbox"/> TESTAUTO.Orders                                       | TESTAUTO                 | TESTAUTO                  | DBMS Table            |

The screenshot shows a Windows-style dialog box titled "Order Number" with a close button (X). The main title is "InfoLibrarian Object Mapping Wizard - Final Step". Below the title, it says "Provide a short description for the mapping and click finish to apply the mappings".

A section titled "The Following mappings/cross references will be added to the object" contains a table with four columns: "Object (Parent.ObjectName)", "Metadata Source", "Metadata Catalog", and "Object Type".

| Object (Parent.ObjectName) | Metadata Source          | Metadata Catalog   | Object Type   |
|----------------------------|--------------------------|--------------------|---------------|
| Northwind.Orders           | SQLServerDemo            | Northwind          | DBMS Table    |
| Northwind.Orders           | Northwind Traders Sam... | Northwind          | DBMS Table    |
| Subject Area Terms.Orders  | Business Terms           | Subject Area Terms | Business Term |

Below the table is a "Mapping Description" text box containing the text "Term Mapping". Below the text box is a line of instructions: "Enter a short description for the mappings and click finished to apply these mappings/cross references". At the bottom of the dialog are four buttons: "< Back", "Next >", "Finish", and "Cancel".

Once you select your mappings, you can assign a description and apply them in a batch. Sometimes databases are replicated or you will be able to determine potential mappings based on the naming of objects. In these cases the Mapping Wizard is the perfect tool for the job.

In the case of replication the description could easily be “One to One Database Copy” or something of this nature.

## 2. The Spreadsheet Template Methodology

The InfoLibrarian Metadata Spreadsheet Templates are an extremely powerful tool for loading, categorizing and building semantics and business metadata.

The Template loader has full access to the repository and scanning API's which allow you to Create Catalogs, Categories, Object types, objects and mappings. You can even perform repository lookups from the spreadsheets.

Having full programmatic control via VBA macros makes the templates easy to setup and maintain while giving you the capability to process, scrub and perform logic while loading your metadata.

Various templates and samples are provided to do the following:

- Create business terms and their mappings
- Define semantics categories and map elements to those categories on load
- Define and map ETL processes
- Create user defined list of reports or other items from sources like the mainframe
- Perform logical calculations of existing metadata to automatically infer mappings in a batch.

The Excel Templates are fully customizable.

## 3. Customization of Scanner Templates (Enterprise Edition Only)

The InfoLibrarian Metadata Scanner Templates can be customized to load, categorize and build semantics and business metadata while scanning source metadata. The scanning templates have full access to the repository and scanning API'S which allow you to Create Catalogs, Categories, Object types, objects and mappings.

Additionally, scanners can be run in workflow or dependency order to build the inter-relationships between toolsets during actual load. Custom scanners could be built to simply generate or cook your subject areas and categories which can then in turn be scheduled.

Implementing scanners in this way fully automated the process and allows the scheduling and execution of the scanning and categorization on the server.

The Enterprise Edition Scanner Templates are fully customizable.

**NOTE:** Excel templates built can be easily ported to production scanners and provide a perfect starting point for building the automation and rules that can then be transformed into full blown scanners.

## Metadata Maintenance and Stewardship

Once you establish the loading processes for capturing your metadata, whether it is technical, business or semantics, Info Librarian's robust scheduling and rescanning architecture allows you to maintain the consistency of your metadata into the future.

With the Enterprise Edition all scanners, including custom scanners are scheduled in production and provides complete automation and capture of changes over time. This includes special mapping scanners and subject area categorizers.

### Maintenance of Daily Ad-hoc Changes Change Management.

In order to properly manage ad-hoc changes to metadata and to keep real-time changes, it is essential to implement the *Change Management and Contribution Portals*. These modules will provide the control, approval and management of changes over time without risking the integrity of your metadata repository.

### Change Management and Real-time History Versioning

The Change Management and Contributor portals are fully web based and through user authentication, changes are posted to change management for approval. Only approved changes are posted to the live metadata repository which insures that you organization will always maintain rich metadata without and concern for the integrity and accuracy of the information.

## Overlapping Scans and Template Loads

In many cases you may find that you will capture some of the metadata via a scan but will need to setup a spreadsheet loader or custom scanner to input additional metadata. Overlapping the scans or loading is acceptable. When objects are recreated via the API, only the deltas or changes are updated. Here is an example:

1. Scan of relational database loads columns.
2. Spreadsheet adds custom properties, mappings and descriptions.
3. Another scanner categorizes the elements from this scan within a subject area category.

## Additional Help

Please refer to the InfoLibrarian Books online, Portal Documentation and other product documentation provided with the product for more information.

Metadata Semantics How To's can be found within the InfoLibrarian Books Online under the section "How To".

Contact [sales@infolibcorp.com](mailto:sales@infolibcorp.com) for more information