

Whitemarsh  
Information Systems Corporation

# Whitemarsh Metabase Resource Life Cycle Analysis Users Guide

November 2007

Whitemarsh Information Systems Corporation  
2008 Althea Lane  
Bowie, Maryland 20716  
Tele: 301-249-1142  
Email: [mmgorman@wiscorp.com](mailto:mmgorman@wiscorp.com)  
Web: [www.wiscorp.com](http://www.wiscorp.com)

## Table of Contents

1	Introduction .....	1
2	Software Installation .....	2
3	Database Design .....	2
4	Reference Data .....	4
5	Operation .....	4
6	Process Model .....	6
6.1	Reference Data .....	7
6.2	Fact Data .....	12
6.2.1	Resource .....	12
6.2.2	Resource Life Cycle Nodes .....	15
6.2.3	Resource Life Cycle Node Structure .....	18
6.3	Assignment Data .....	21
6.3.1	Resource Life Cycle Node Mission Assignments .....	21
6.3.2	Resource Life Cycle Node Information Need Assignments .....	21
6.3.3	Resource Life Cycle Node Database Object Assignments .....	21
6.3.4	Resource Life Cycle Node Business Information Systems Assignments .....	21
6.4	Reports .....	26



## List of Figures

<b>Figure 1.</b> Resource Life Cycle Data Model. ....	3
<b>Figure 2.</b> Login screen. ....	5
<b>Figure 3.</b> Resource Type list. ....	7
<b>Figure 4.</b> Resource Type update screen. ....	9
<b>Figure 5.</b> Resource Life Cycle Node Structure Type list. ....	10
<b>Figure 6.</b> Resource Life Cycle Node Structure Type update screen. ....	11
<b>Figure 7.</b> Resource list. ....	12
<b>Figure 8.</b> Resource update screen. ....	14
<b>Figure 9.</b> Resource Life Cycle Node list. ....	15
<b>Figure 10.</b> Resource Life Cycle Node update screen. ....	17
<b>Figure 11.</b> Resource Life Cycle Node Structures list. ....	19
<b>Figure 12.</b> Resource Life Cycle Node Structure update screen. ....	20
<b>Figure 13.</b> Mission Resource assignments. ....	22
<b>Figure 14.</b> Resource Life Cycle Node Information Need assignment. ....	23
<b>Figure 15.</b> Resource Life Cycle Node Database Object assignment. ....	24
<b>Figure 16.</b> Resource Life Cycle Node Business Information System assignment. ....	25



## 1 Introduction

The purpose of the database application, Resource Life Cycle Analysis is to build resource life cycle networks, that is, the resource, life cycle and the precedence vectors between resource life cycles. A **resource** is an enduring asset of value to the enterprise. The **life cycle** is a linear identification of the major states that must exist within life of the resource. The life cycle of a resource represents the resource's "cradle to grave" set of state changes. The **precedence** is a vector that may occur between nodes on different resource life cycles, and thus indicates which resource life cycle node enables another resource life cycle node.

Resource life cycles are a key component in the planning, development, and evolution of Enterprise Database. Resource Life Cycles are properly identified subsequent to mission and database object analysis, and once created serve as a framework for understanding how all the databases and information systems from Information Technology serve the business. The RLC network gives IT organizations the ability to answer questions such as:

- What IT projects address a particular segment of the enterprise?
- What business resource is served/enabled when one project be started and completed before another?
- What IT projects are largely redundant one with another?
- Which segments of the RLC networks are either under or over served by IT?

Resource life cycle networks are an important component of information systems planning. The detailed methodology for building RLC networks is contained in the first phase of the Whitemarsh methodology.

### Presumed Knowledge

This user guide, and all the other metabase user guides presume that the reader has read and is completely familiar with the following documents: Metabase Common Processes, and Metabase Bill of Materials and Single File Recursion (BOM/SFR Guide). These two documents serve as metabase teaching guides for processes that commonly occur throughout the metabase system.

F7 invokes automatic spell checking on all text fields like names and descriptions.

### Metabase Example

The metabase example, Movies, is a complete example of a business which is available from the Whitemarsh website. The Movies Rental Corporation was modeled after the largest movies rental corporation in the United States. As such, the example has national, regional, and retail



outlets. There are two data models, one for an original data capture, store based system, and another which is a data warehouse for rented movies.

## 2 Software Installation

Metabase installation is explained in the Metabase Administrator's Guide.

## 3 Database Design

The database consists of nine different tables in this database application. The tables are:

- Mission Resources
- Resource
- Resource Life Cycle Node Structure
- Resource Life Cycle Node
- Resource Life Cycle Node Structure Type
- Resource Life Cycle Node Information Need Assignment
- Resource Life Cycle Node Database Object Assignment
- Resource Life Cycle Node Business Information Systems Assignment
- Resource Type

There are 6 additional entities in the database. These are:

- Business Information System
- Database
- DBMS
- Information Need
- Information Need Type
- Mission

These entities are shaded because they are read-only within this module. They are created/updated in the Database, Business Information Systems, Information Needs Analysis, and Mission-Organization-Function-Person metabase modules.

The database design is depicted in Figure 1. Explicit in this database design are the following:

**Resource Life Cycle Node Business Information Systems Assignment.** A Resource Life Cycle Node Business Information Systems Assignment is an association between a resource life cycle node and a business information system. A business information system may be assigned to one or more resource life cycle nodes and a resource life cycle node may be related to one or more business information systems.



All Rights Reserved  
02/18/2000



**Resource Life Cycle Node.** A Resource Life Cycle Node is a life cycle state within the resource. If the resource is employee then the life cycle node may be employee requisition, employee candidate, employee new hire, assigned employee, reviewed employee, and separated employee.

**Resource Life Cycle Node Structure Type.** A Resource Life Cycle Node Structure Type is a classification of a set of resource life cycle node structures that explain the collection. An example might be enablement and the associated resource life cycle node structures might related a recognized receivable resource life cycle node from the receivables resource “enables” a paid invoice resource life cycle node from the invoice resource.

**Resource Type.** A Resource Type represents a collection of resources. For example, finance resource resources, or property resources.

**Resource.** A Resource is an enduring asset of value to the enterprise. Included for example are facilities, assets, staffs, money, even abstract concepts like reputation. If a resource is missing then the enterprise is incomplete.

## 4 Reference Data

The two reference data tables are:

- Resource type
- Resource Life Cycle Node Structure Type

The values employed in resource type enable the broad classification of business domains withing which resource life cycles operate. Typical values include, Finance, and Human Resources.

Resource Life Cycle Node Structure Types serve to classify the various bills-of-materials data that is present in the resource life cycles.

## 5 Operation

Once the application is installed it is ready to use. Just invoke the software from the metabase program. The application is a traditional windows application. Metabase reports are accomplished through any ODBC class report writer such as Crystal Reports.



## 5.1 Log In Process

Figure 2 shows the log-in screen that appears immediately after the application is started. Entered is your user name and your password. These are created by the Metabase Administrator through the metabase administration module. Please contact your metabase administrator to set up your user name and password. Once a user name and password is established, all the user's information can be changed by the user through a restricted use version of the administrator software. Once the send button is pressed the specific metabase database instances that can be accessed by the user is presented. The metabase is such that users are allowed to use specific metabase instances and specific metabase modules.

Resource Lifecycle Analysis Component - Whitemarsh Metabase

File Edit ResourceLifecycle Assignments ReferenceData Window Help

Current Dataset

Dataset **Movies**

Select a Metabase Data Source Node

Step 1) Enter Username & Password

Username: MyUserName Password: MyPassword Send ...

Step 2) Select a Dataset

DSN	Description
Movies	Movies Metabase

Select Close

**Figure 2.** Login screen.





In this particular example, the user, once they sent their user name and password are shown the metabase database that they can access, that is, Movies. Highlight the choice and press the Select button. Once that is done then the metabase name, Movies, is shown as the data set that is being accessed.

## 6 Process Model

The information needs analysis process model consists of four classes of processes:

- Reference Data
- Resource Life Cycle Fact Data
- Assignments
- Reports

If there is an attempt to perform a Delete operation but there are existing records that are dependent on that instance, the delete operation is refused. The complete menu for this module is presented in the table that follows.

<ul style="list-style-type: none"><li>-- Resource Lifecycle<ul style="list-style-type: none"><li>-- Resources</li></ul></li><li>-- Resource Life Cycle Nodes<ul style="list-style-type: none"><li>-- Resource Lifecycle Nodes</li><li>-- Resource Lifecycle Node Structures</li><li>-- Resource Lifecycle Node Structure Types</li></ul></li><li>-- Assignments<ul style="list-style-type: none"><li>-- Assign Mission to Resources</li><li>-- Assign Resource Lifecycle Nodes to Information Needs</li><li>-- Assign Resource Lifecycle Nodes to Database Objects</li><li>-- Assign Resource Lifecycle Nodes to Business Information Systems</li></ul></li><li>-- Reference Data<ul style="list-style-type: none"><li>-- Resource Types</li></ul></li></ul>
<b>Menu for Resource Life Cycle Analysis</b>



## 6.1 Reference Data

The RLC reference data consists of two types:

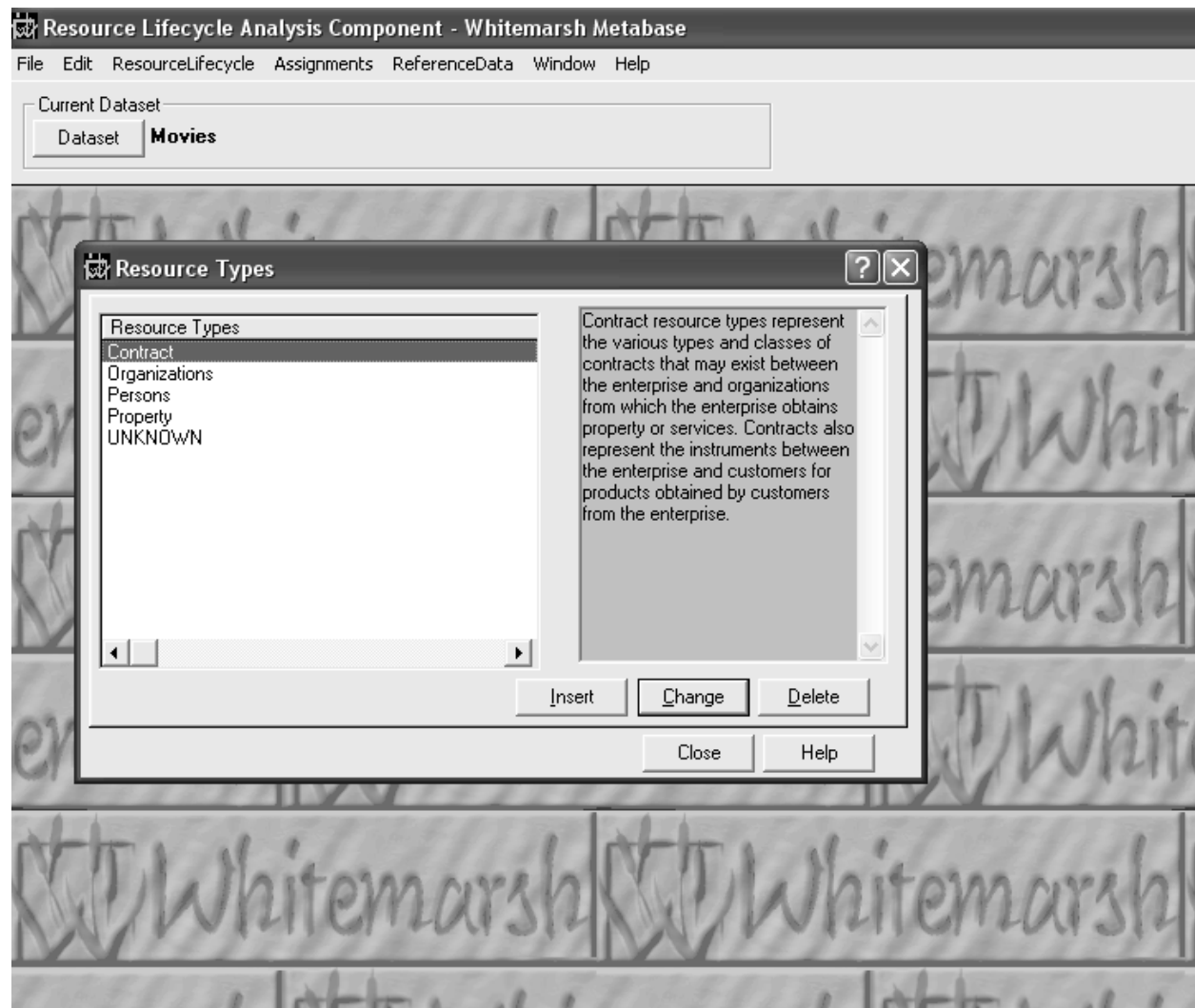
- Resource Type
- Resource Life Cycle Node Structure Type

The RLC reference data is accessible through the Reference Data Menu. Figure 3 presents a browse list screen superimposed by a change screen for Resource Type. The information that can be inserted or changed is the name of the Resource Type and its description.

Resource Life Cycle Node Structure Type update window is presented in Figure 4. This window allows for the entry of three fields: Active-phrase-phrase, passive phrase, and rationale for the particular bill-of-materials. When a precedence vector is created between two different nodes of two resource life cycles, the rationale for the vector must be established. The following table provides examples of precedence vector types and their rationales.

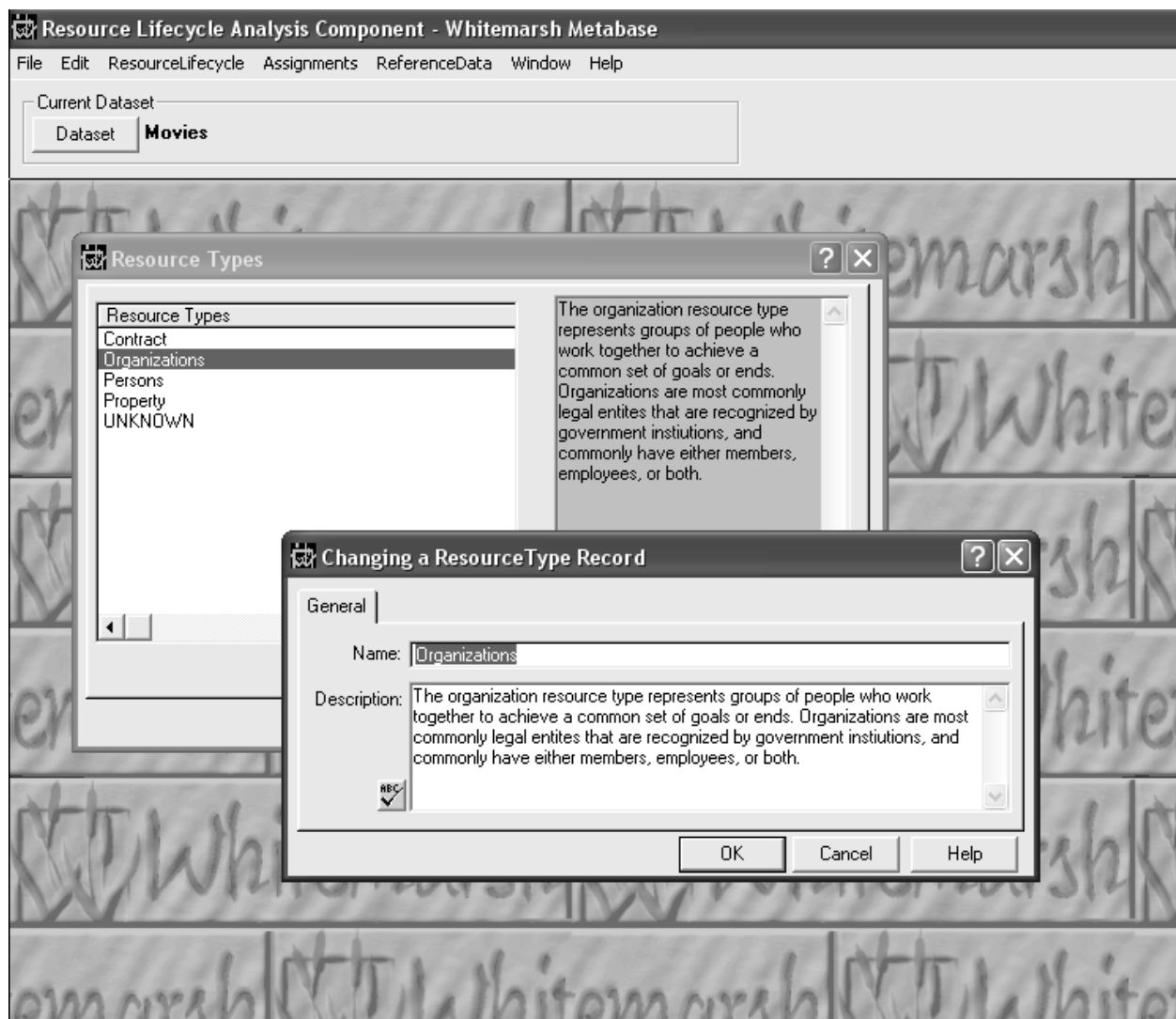
Active phrase	Passive phrase	Rationale
enables	is enabled by	the active resource must be in place so that the passive resource can effectively function
necessitates	is necessitated by	the active resource is essential to exist so that the passive resource can be accomplished at all
authorizes	is authorized by	the active resource is legally required as an authorizing mechanism before the passive resource can be legally done





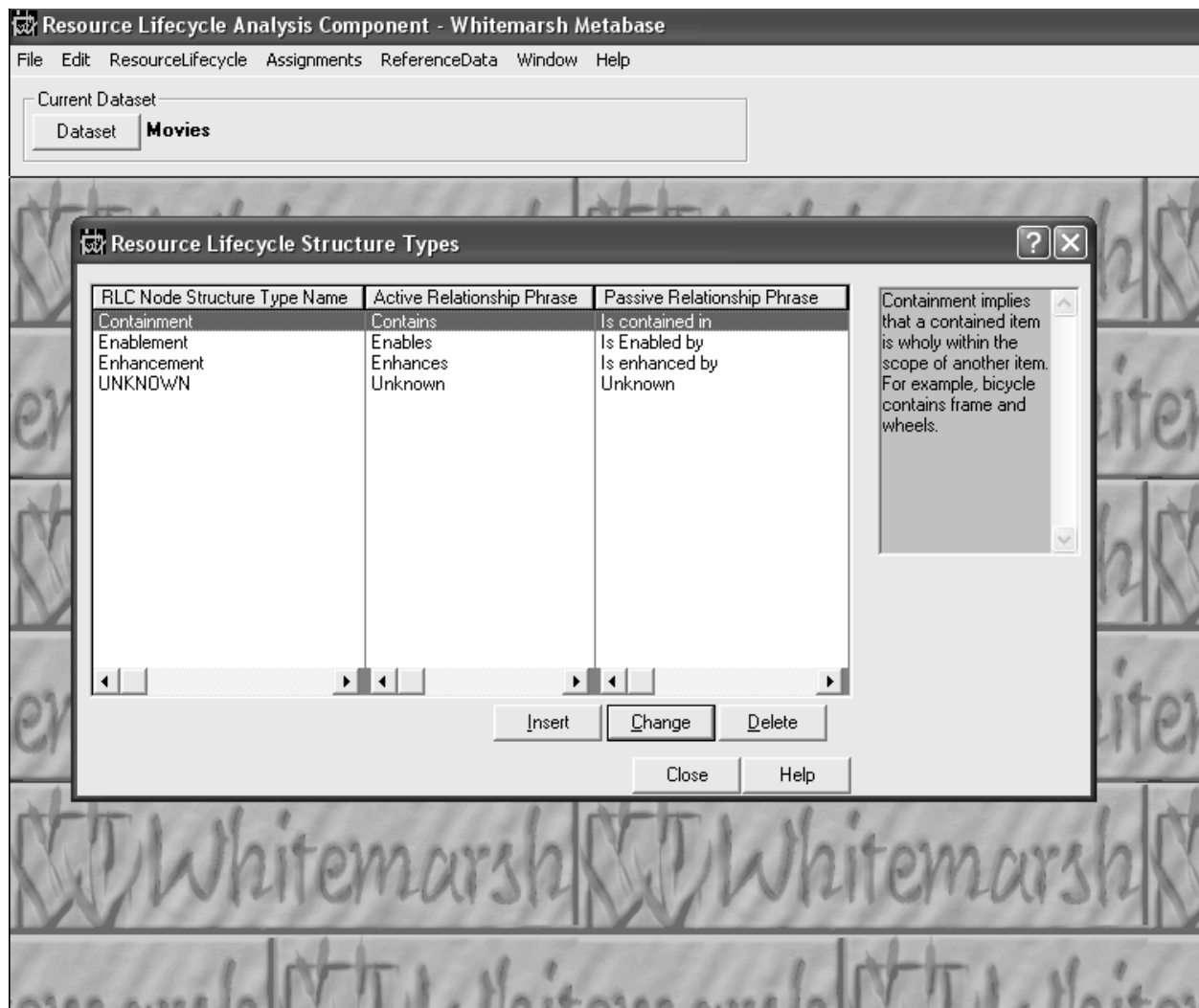
**Figure 3.** Resource Type list.





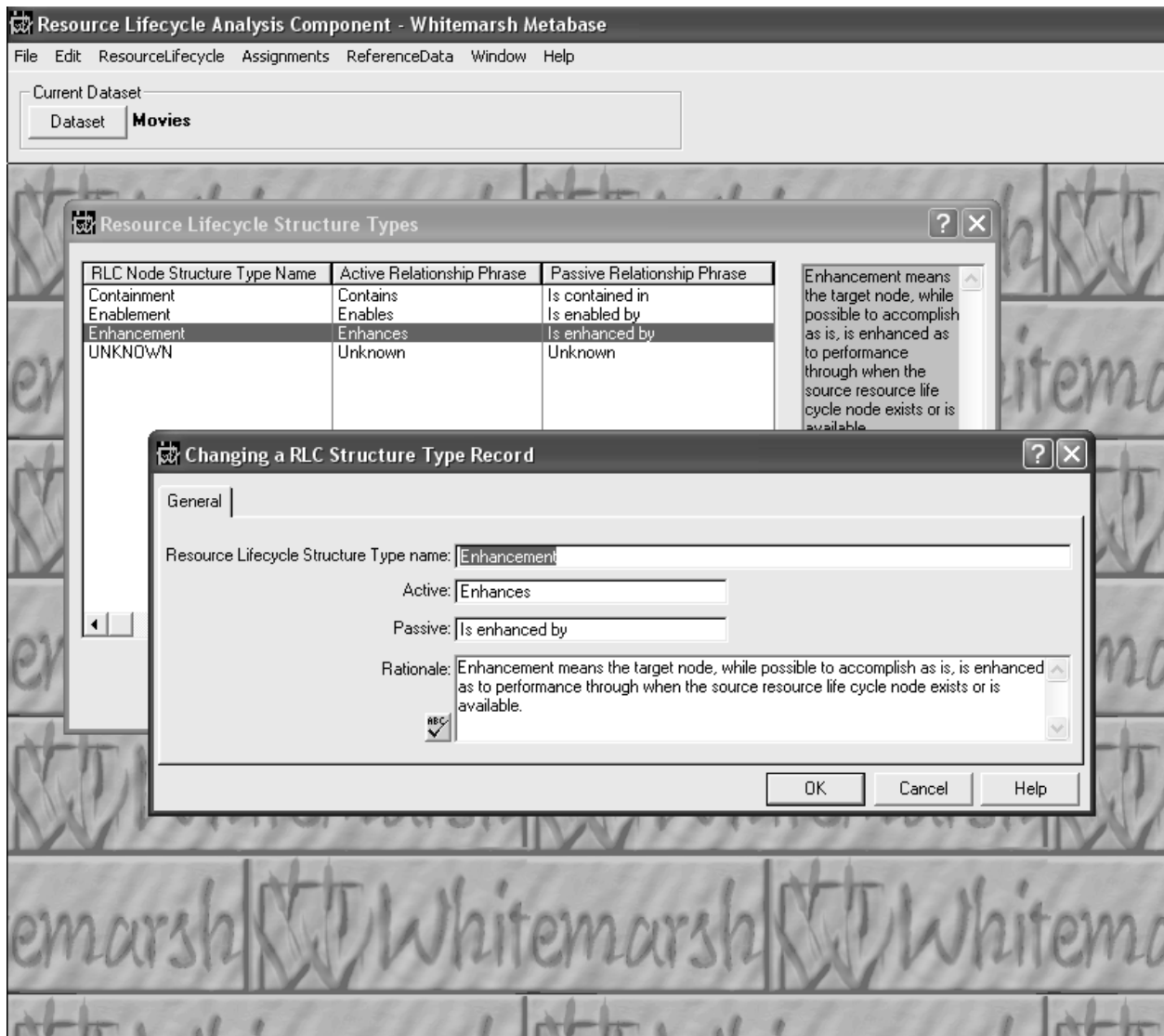
**Figure 4.** Resource Type update screen.





**Figure 5.** Resource Life Cycle Node Structure Type list.





**Figure 6.** Resource Life Cycle Node Structure Type update screen.



## 6.2 Fact Data

The fact data process consists of data for the following:

- Resource
- Resource Life Cycle Node
- Resource Life Cycle Node Structure

### 6.2.1 Resource

Figure 7 presents a browse for resource. Resources can be hierarchical. If a resource has contained resources, select the parent, for example, vendor, and press insert. In the vendor example there are three contained resources, Services, Software, and Hardware.

Figure 8 presents the update screen for resource which allows entry of three fields. The first is the resource type. This is accomplished by putting a Zero in that field and the pressing Tab. Since Zero is an invalid value, the list of existing resource types is presented. Select the appropriate one and the press the Select button. After the resource type is selected, the resource's name is entered and then a description is provided.



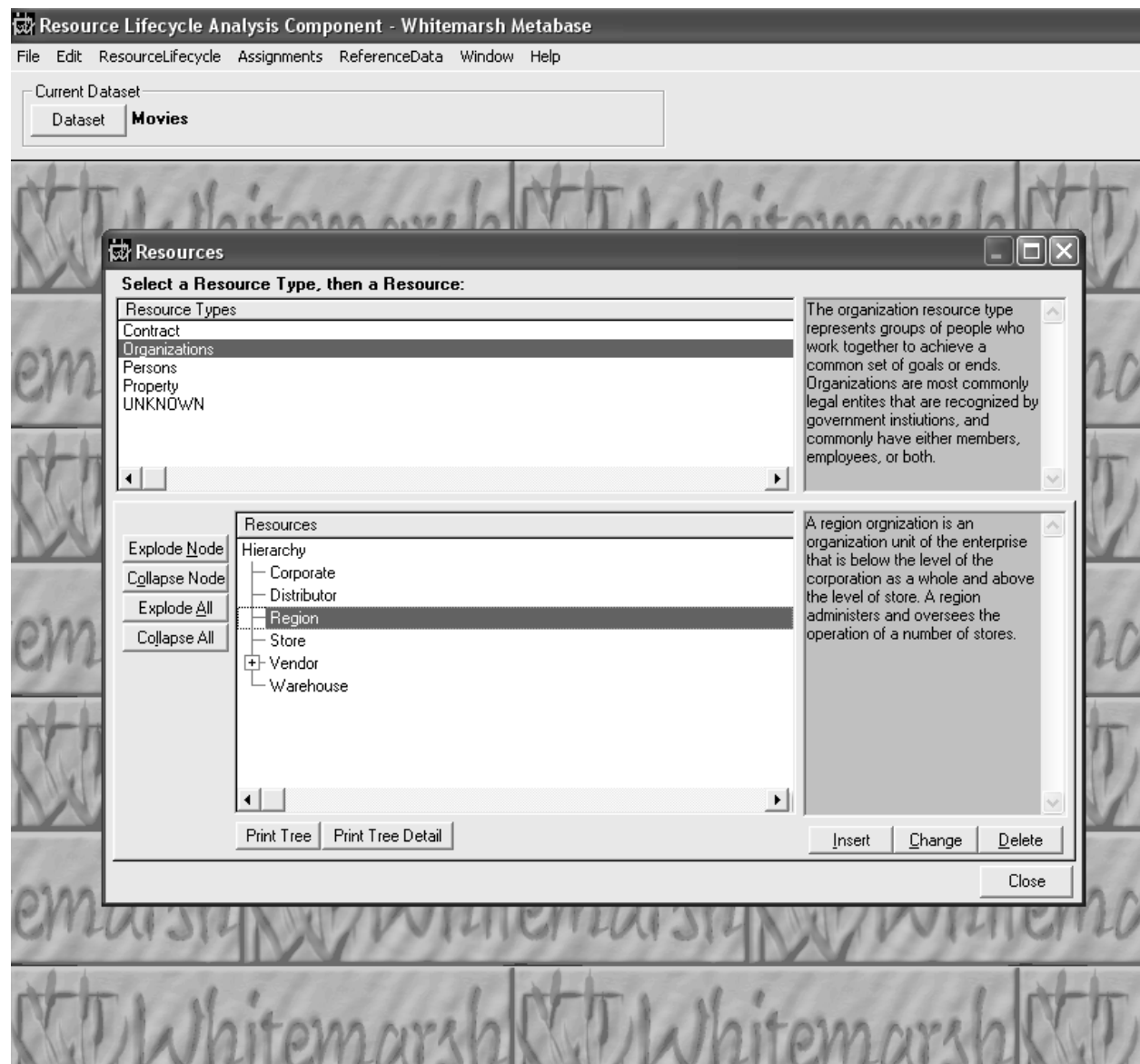


Figure 7. Resource list.





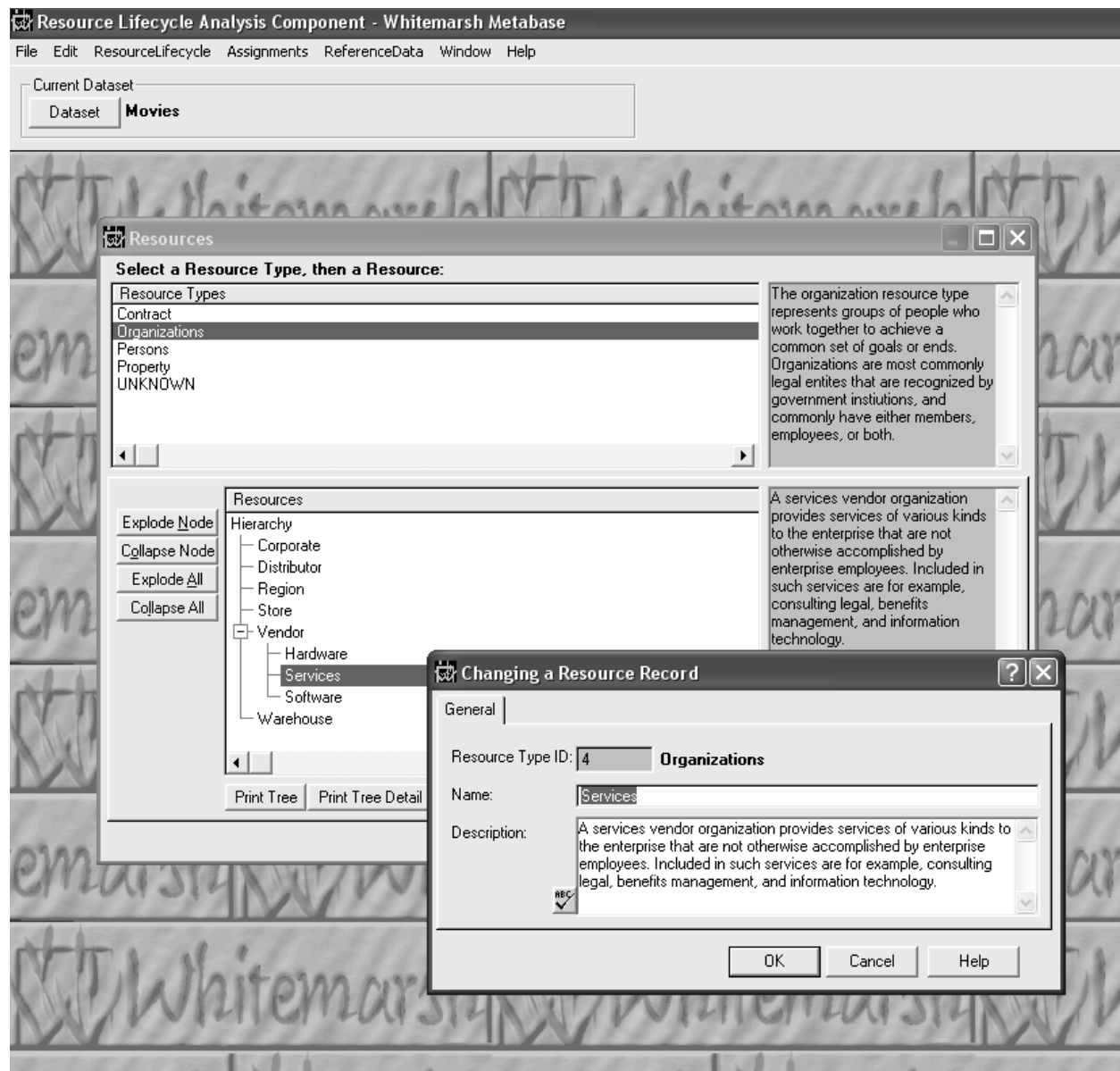
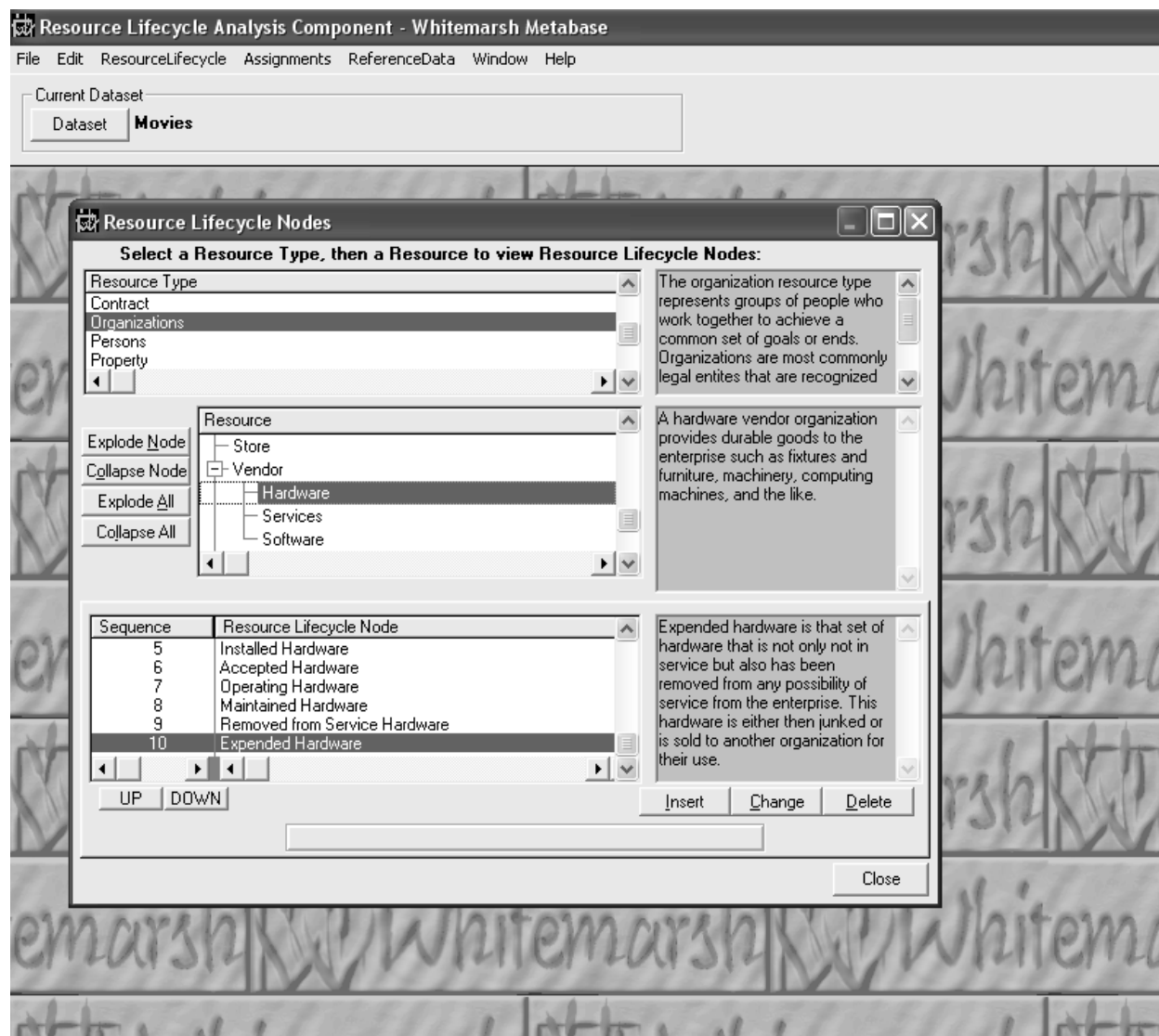


Figure 8. Resource update screen.



## 6.2.2 Resource Life Cycle Nodes

Each resource can then have a life cycle. Each distinct item in the life cycle is a node. Hence, the names for these nodes, Resource Life Cycle Nodes. Figure 9 presents a list for the resource life cycle nodes for each resource.



**Figure 9.** Resource Life Cycle Node list.



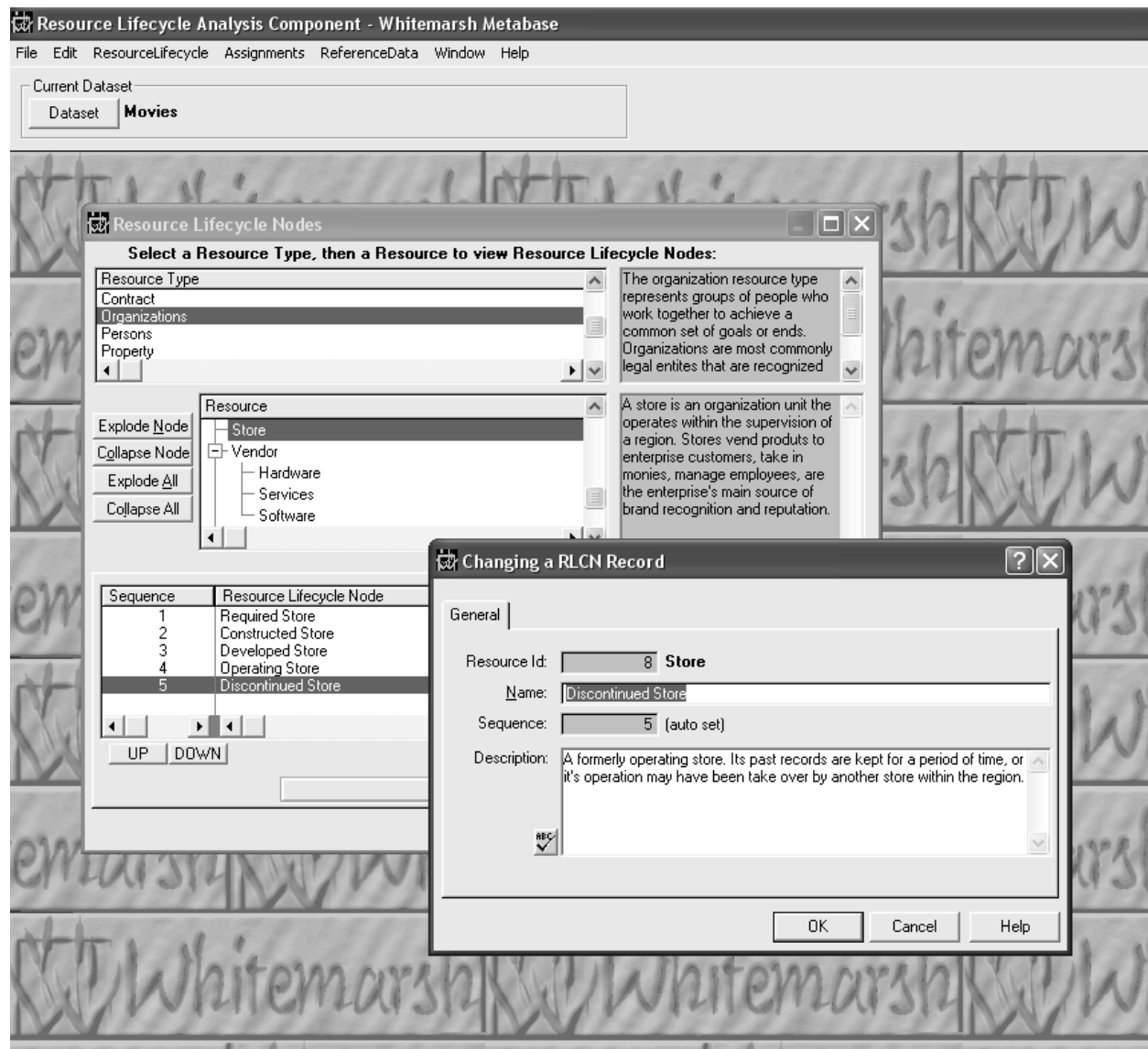
In this particular example, the resource, Vendor has three subordinate resources, Hardware, Services, and Software. The Resource Life Cycle has 10 nodes as follows:

1. Required Hardware
2. Specified Hardware Requirements
3. Selected Hardware
4. Procured Hardware
5. Installed Hardware
6. Accepted Hardware
7. Operating Hardware
8. Maintained Hardware
9. Removed from Service Hardware
10. Expended Hardware

Resource life cycle nodes cannot have subordinate life cycle nodes. That is an unnecessary level of detail. But, Resources can have lower levels. So, the resource Vendor, which already has a subordinate resource, software, might in turn have subordinate levels of Business Software and Systems Software. And in turn, Business Software could have subordinate levels of Package, or Generated, or Custom. And finally, Custom would then have a life cycle appropriate for the key life cycle nodes of custom business software. In short, there's plenty of room for subordinate resources and then a single life cycle for each resource leaf node.

Figure 10 shows an update screen for a resource. In this case, the resource type, Organizations is selected, then store, then the Resource Life Cycle Node, Discontinued. The change button is pressed and you can change its name and/or description. If a new node is to be inserted, add a name and description. As you tab-through the Sequence field it will auto-number. Back on the list screen, Figure 9, you can adjust the sequence numbers through the use of the Up and Down buttons.





**Figure 10.** Resource Life Cycle Node update screen.



### 6.2.3 Resource Life Cycle Node Structure

Figure 11 presents a browse window of Resource Life Cycle Node Structures. In this window the top browse contains the Resource Life Cycle Structure Types. Select one and the Resource Life Cycle Structures are subset accordingly. In this example, Enablement was selected. In the Resource Life Cycle Node browse, the top level structures shown are Operating Store, and Required Corporate Organization. Within Operating Store is Employee Requirement, and Candidate Customer. Within Required Corporate Organization is Proposed Distributor, Required Region, which in turn has Required Store, and Required Warehouse.

A review of the Resource Life Cycles for the Resources shows the complete life cycle for each. The connections here are at the first node of the Resource's Life Cycle. Thus, what is being stated here is that once a store is operating there can be employees and there can be customers. Similarly, the Required Corporate Organizations start with Proposed Distributors, Required Regions, and Required Warehouses. The source resource (Corporate Organization) and the appropriate Resource Life Cycle Node (Required Corporate Organization), once in place and operating appropriately then enables (Enablement) the target resources (other Corporate Organizations) such as Distributors, Regions, and Warehouses.

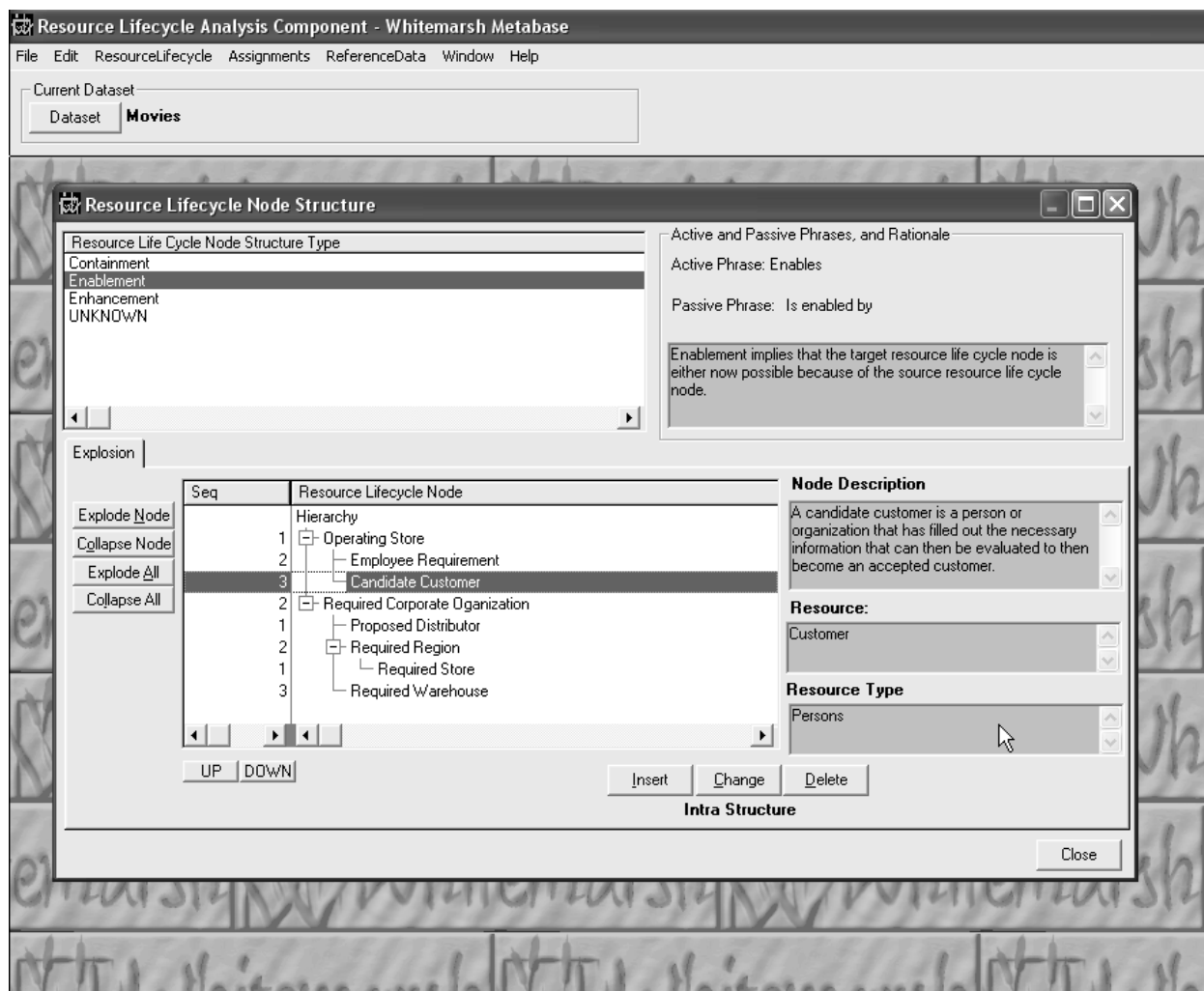
To the right of both the top and bottom browses are text description blocks.

Figure 12 presents the update window for inserting, changing, or deleting resource life cycle node structures. This screen consists of three parts. The top part is for the Enabling (source) Resource and Resource Life Cycle Node. The middle part is for the basis of the relationship and the third part is for the Enabled Resource and Resource Life Cycle Node. In this example, the relationship effectively states that:

Section of Screen	Meaning of Section
Top (enabling)	Operating Store
Middle (reason type)	enables
Bottom (enabled)	Proposed Rental Agreements

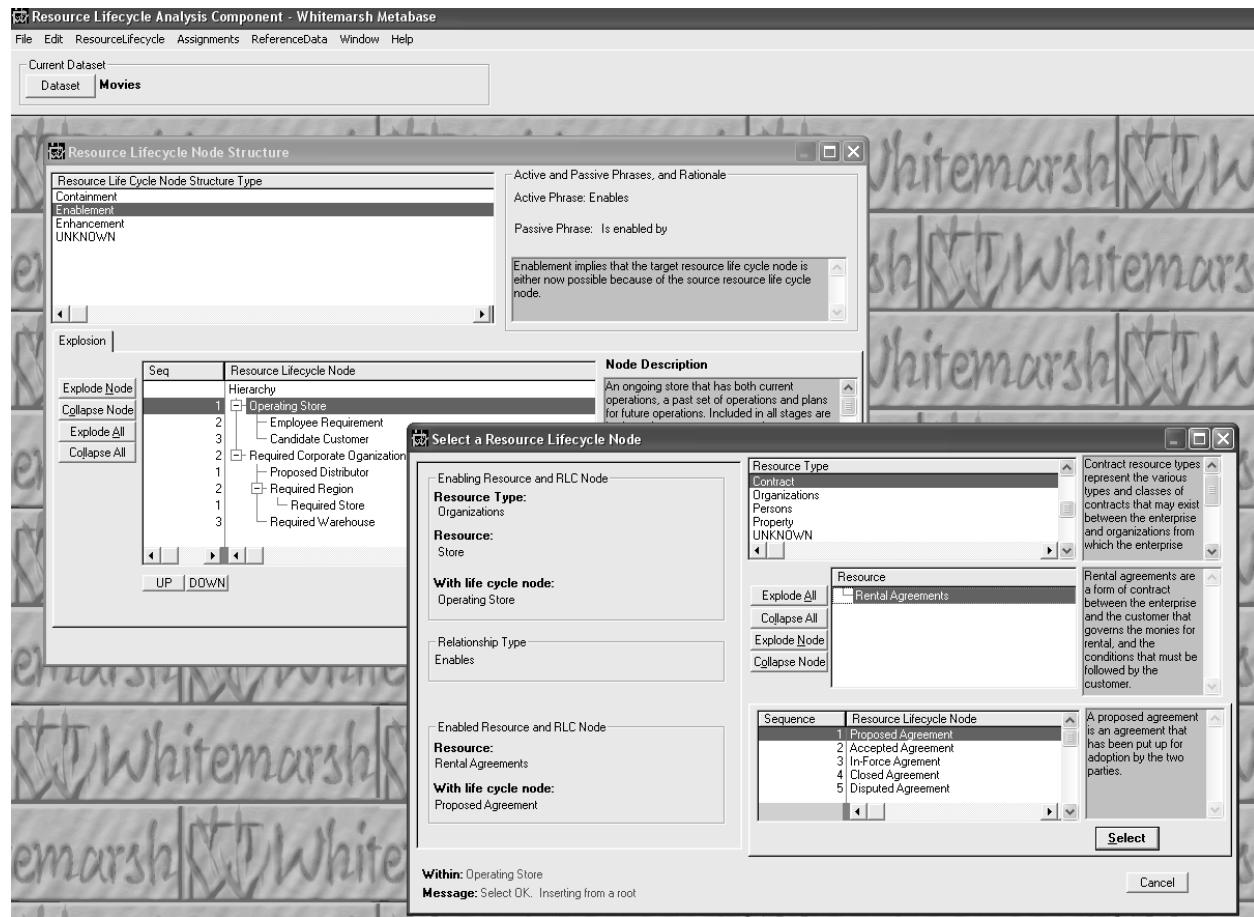
The enabling Resource Life Cycle Node is selected on Figure 11. The insert button is pressed. Then the update screen as shown in Figure 12 is presented. Select the Resource Type, then the Resource, then (most commonly) the first Resource Life Cycle Node. Then press the Select button. When the screen closes, the selected Resource Life Cycle Node will appear as a child of the one selected via Figure 11.





**Figure 11.** Resource Life Cycle Node Structures list.





**Figure 12.** Resource Life Cycle Node Structure update screen.



## 6.3 Assignment Data

The assignment data processes consist of four types:

- Resource Life Cycle Mission Assignments
- Resource Life Cycle Node Information Need Assignments
- Resource Life Cycle Node Database Assignments
- Resource Life Cycle Node Business Information Systems Assignment

### 6.3.1 Resource Life Cycle Node Mission Assignments

Figure 13 presents an intersection record operation for relating Resources and Missions. The strategy is to tag one mission and then tag as many different resources are involved in dealing with that mission. Once all the appropriate organizations are tagged, the Build button is pressed. After a short while the intersection records, that is, those resources identified with the missions appear in the bottom window.

### 6.3.2 Resource Life Cycle Node Information Need Assignments

Figure 14 presents an intersection record operation for relating Resource Life Cycle Nodes and Information Needs. The strategy is to tag one Resource Life Cycle Node and then tag as many different Information Needs are involved in dealing with that Resource Life Cycle Node. Once all the appropriate Information Needs are tagged, the Build button is pressed. After a short while the intersection records, that is, those Information Needs identified with a particular Resource Life Cycle Node appear in the bottom window.

### 6.3.3 Resource Life Cycle Node Database Object Assignments

Figure 15 presents an intersection record operation for relating Resource Life Cycle Nodes and Database Objects. The strategy is to tag one Resource Life Cycle Node and then tag as many different Database Objects are involved in dealing with that Resource Life Cycle Node. Once all the appropriate Database Objects are tagged, the Build button is pressed. After a short while the intersection records, that is, those Database Objects identified with a particular Resource Life Cycle Node appear in the bottom window.

### 6.3.4 Resource Life Cycle Node Business Information Systems Assignments

Figure 16 presents an intersection record operation for relating Resource Life Cycle Nodes and Business Information Systems. The strategy is to tag one Resource Life Cycle Node and then tag





as many different Business Information Systems are involved in dealing with that Resource Life Cycle Node. Once all the appropriate Business Information Systems are tagged, the Build button is pressed. After a short while the intersection records, that is, those Business Information Systems identified with a particular Resource Life Cycle Node appear in the bottom window.

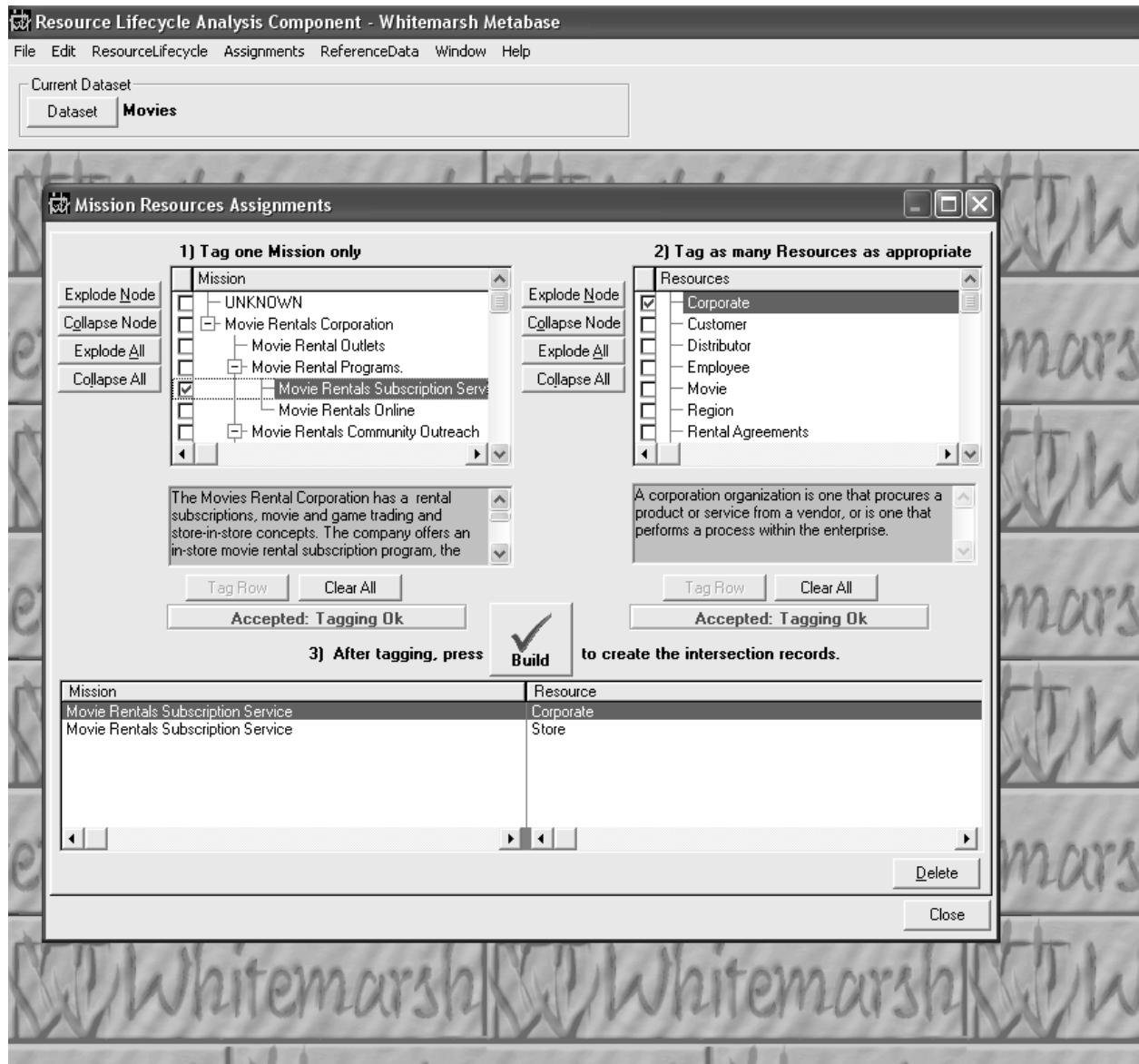


Figure 13. Mission Resource assignments.



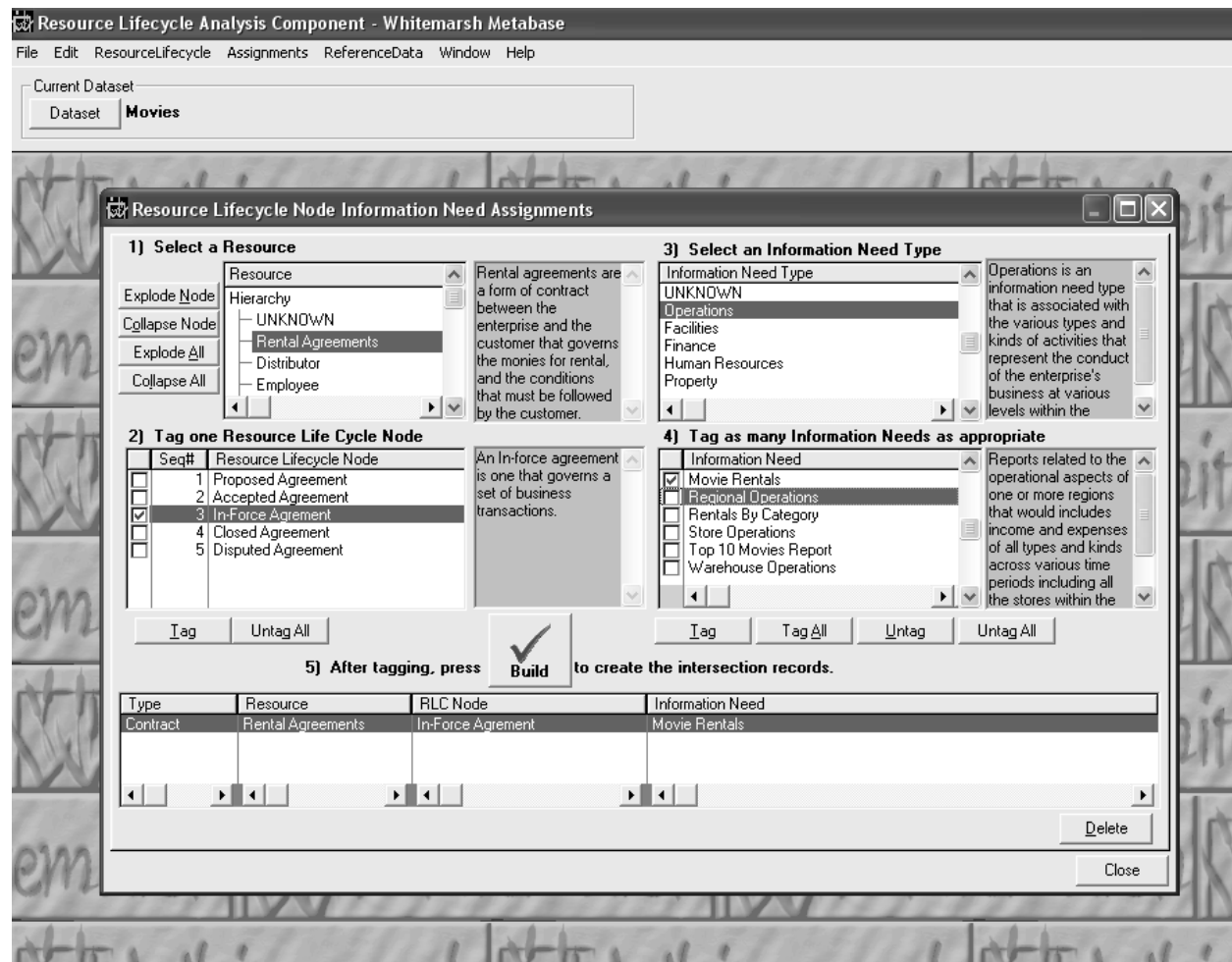


Figure 14. Resource Life Cycle Node Information Need assignment.



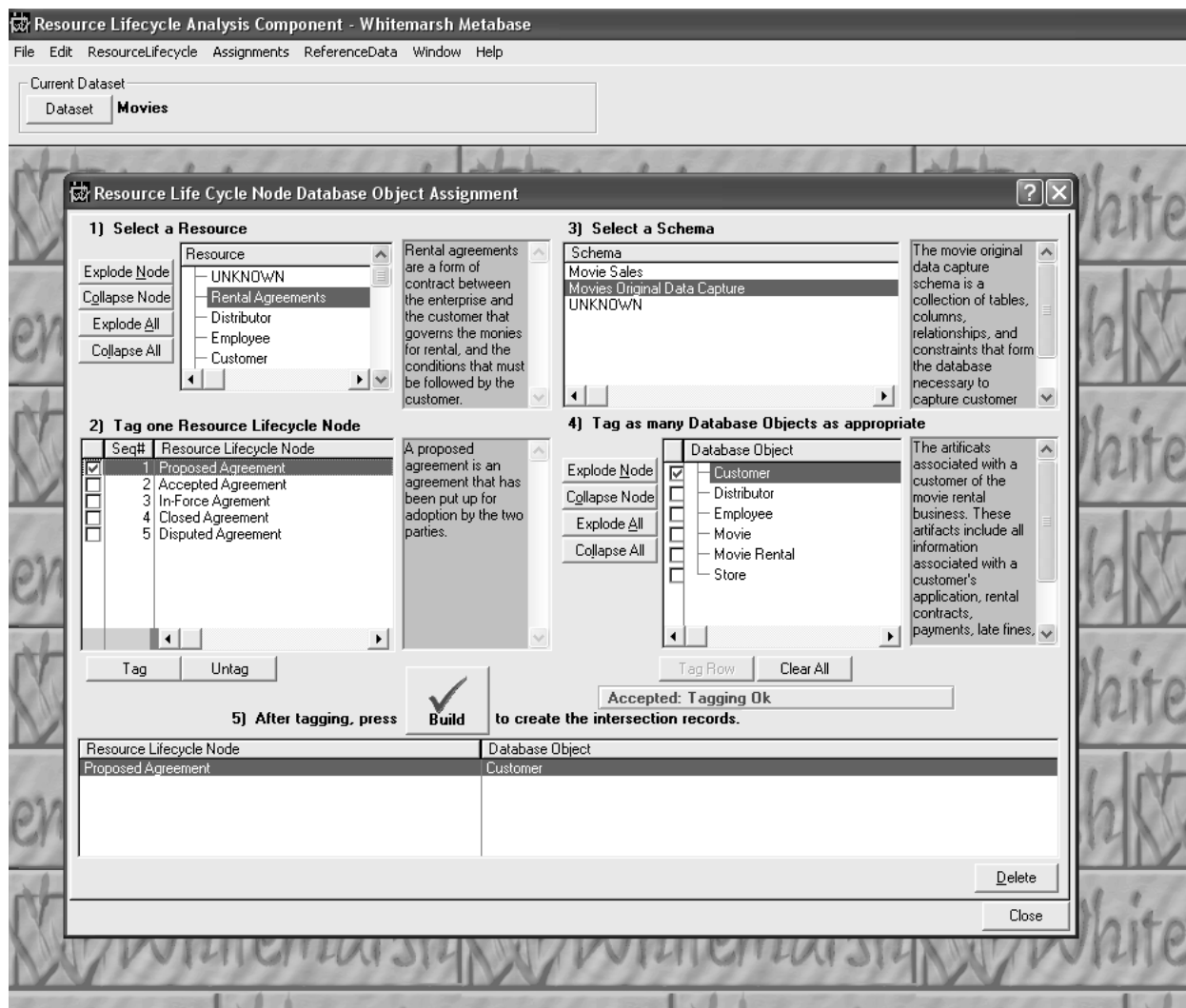
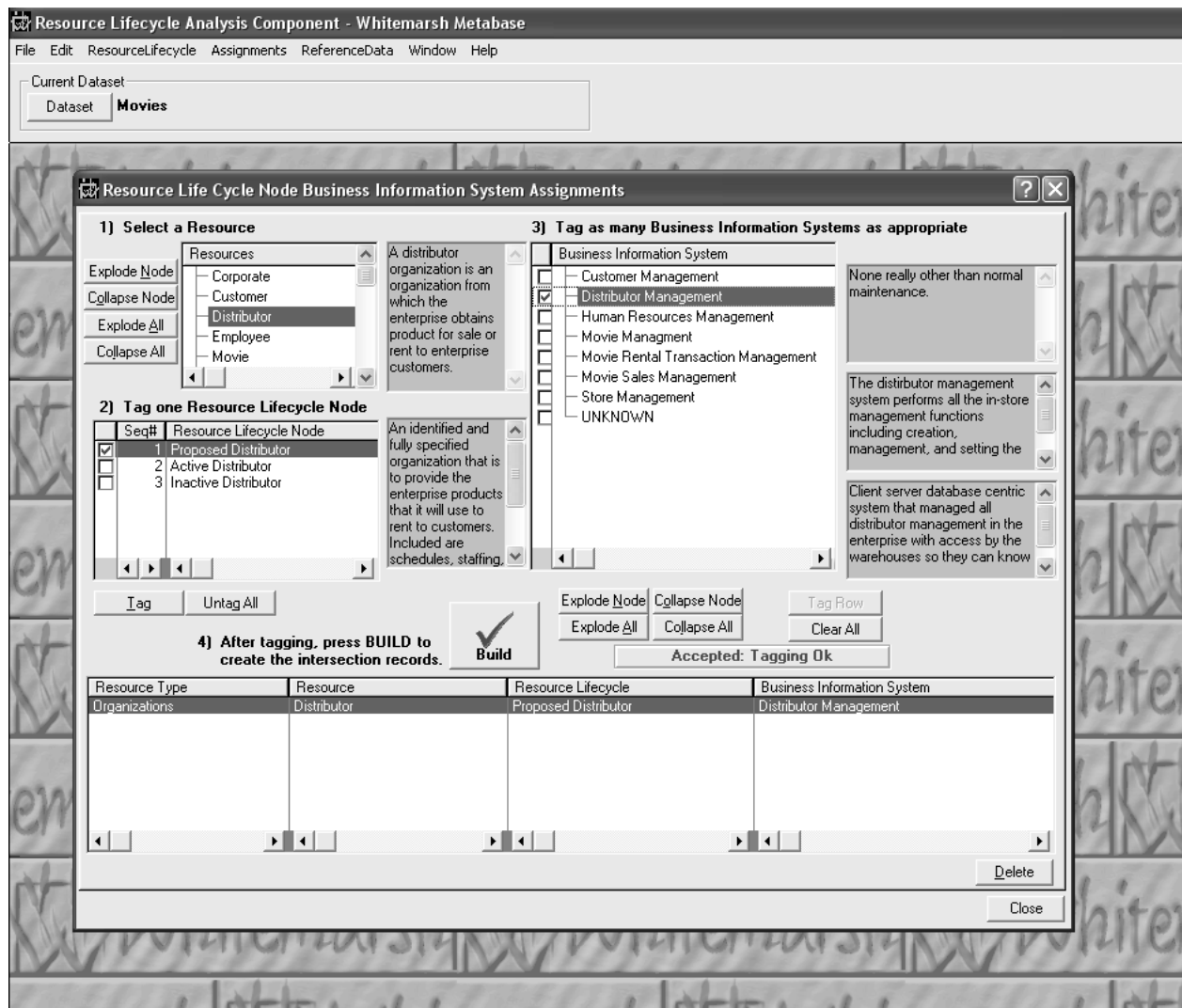


Figure 15. Resource Life Cycle Node Database Object assignment.





**Figure 16.** Resource Life Cycle Node Business Information System assignment.



## **6.4 Reports**

Reports are accomplished through access to a particular metabase database instance through commercial report writers such as Crystal Reports. Whitemarsh provides about 100 such report templates for Crystal Report access from the Whitemarsh website.

