

## About OpenTravel:

The OpenTravel Alliance provides a community where companies in the electronic distribution supply chain work together to create an accepted structure for electronic messages, enabling suppliers and distributors to speak the same interoperability language, trading partner to trading partner. Tens of thousands of the OpenTravel message structures are in use, carrying tens of millions of messages between trading partners every day.

Members do the work of identifying what messages are needed, prioritize the work and collaborate to create the messages. Members who are looking for more information on related project team work or who wish to access the OTM repository can send inquiries to [architecture@opentravel.org](mailto:architecture@opentravel.org).

**Note:** This document supports implementers using the OTM-DE Model Builder in the creation and sharing of models that automatically generate xml schema. The ability to access and extend the OpenTravel Model is available only to OpenTravel members. For more information please contact us at [membership@opentravel.org](mailto:membership@opentravel.org).

# OTM-DE Repository User Guide

## Document Purpose:

**This document explains the distributed architecture of repositories that you set up and manage as part of the OpenTravel Model Development Environment (OTM-DE).**

## Contents

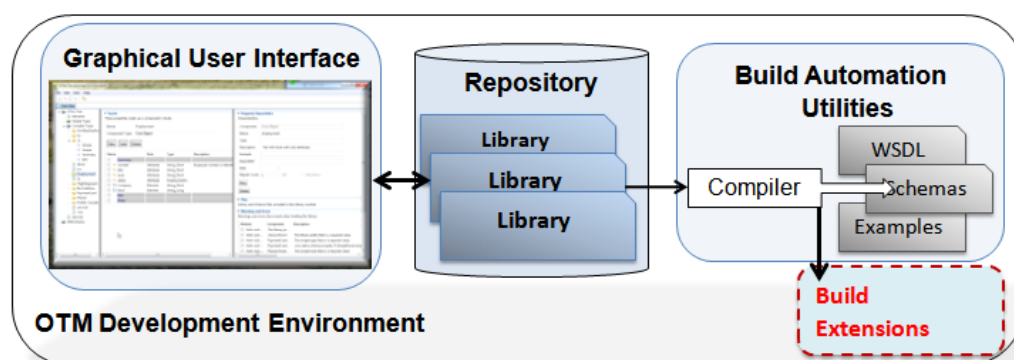
Overview .....	2
OTM Repository Architecture .....	2
Repository Access Controls.....	3
Namespace Structure.....	4
Projects and Libraries.....	5
Users and Groups.....	6
Permissions .....	6
Additional Access Controls.....	6
Status Bar .....	7
Task 1 — Proxy Configuration.....	7
Task 2 — Access a Repository .....	8
Summary .....	10

Task 3 — Search .....	10
Summary .....	10
Task 4 — Accessing Libraries .....	11
Task 5 — Edit Library.....	11

## Working with OTM Repositories

### Overview

The OTM Development Environment includes OpenTravel Model (OTM) Repositories. The OTM Repository is custom designed to manage and share OTM Libraries and fully integrate with the OTM Development Environment

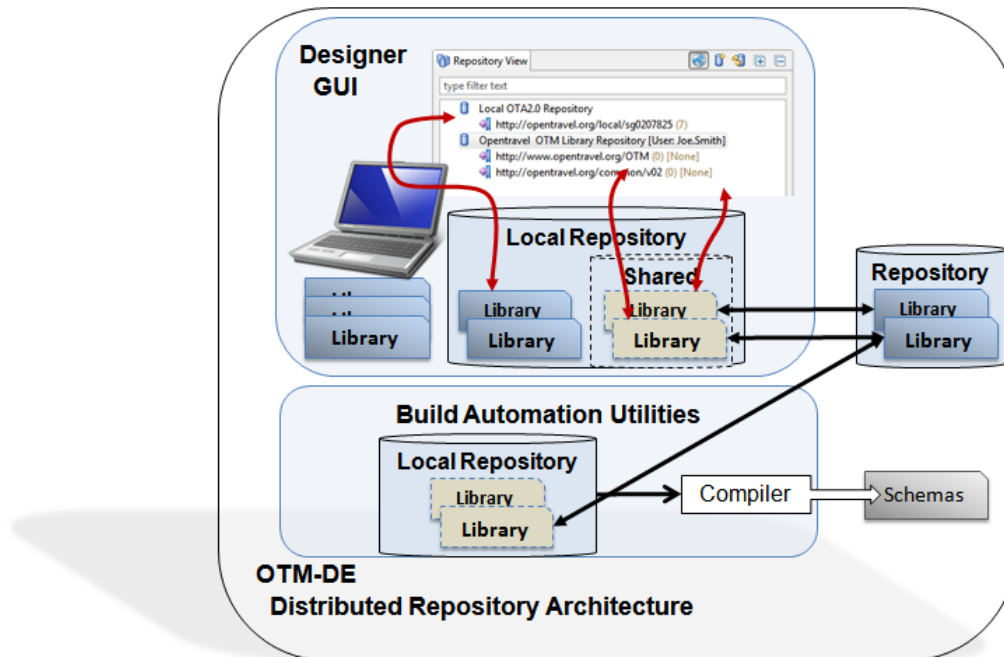


The OTM Repository implements the basic functionality of a repository of storing and accessing OTM Libraries. In addition the OTM-DE Repository has a variety of OTM specific functions including search, access controls, namespace management, versioning, state management and locking.

This user guide provides an introduction to the OTM Repository, namespaces and *access controls*. The guide then describes how users interact with the *Repository* and *Access Controls* using OTM Projects, Libraries and Namespaces. The guide covers both the repository web interface and the OTM-DE Model Designer user interface. This guide assumes the reader is familiar with the concepts described in [OTM-DE Namespaces Reference](#).

### OTM Repository Architecture

Repositories are designed with a *distributed architecture*. The *Model Designer* user interface (GUI) and Build Automation Utilities automatically install a *local* repository. The distributed design allows library designers to access multiple repositories, improves performance, and provides access to managed libraries even when there is no connectivity.



**Local Repository:** The distributed repository architecture includes a *local repository*. The GUI and Build Automation Utilities deploy a local repository that uses the local file system for storage. The GUI's Repository View automatically connects to this repository.

The local repository also provides storage for copies of libraries shared from other repositories. The shared libraries feature assures access to remote libraries even when connectivity is limited.

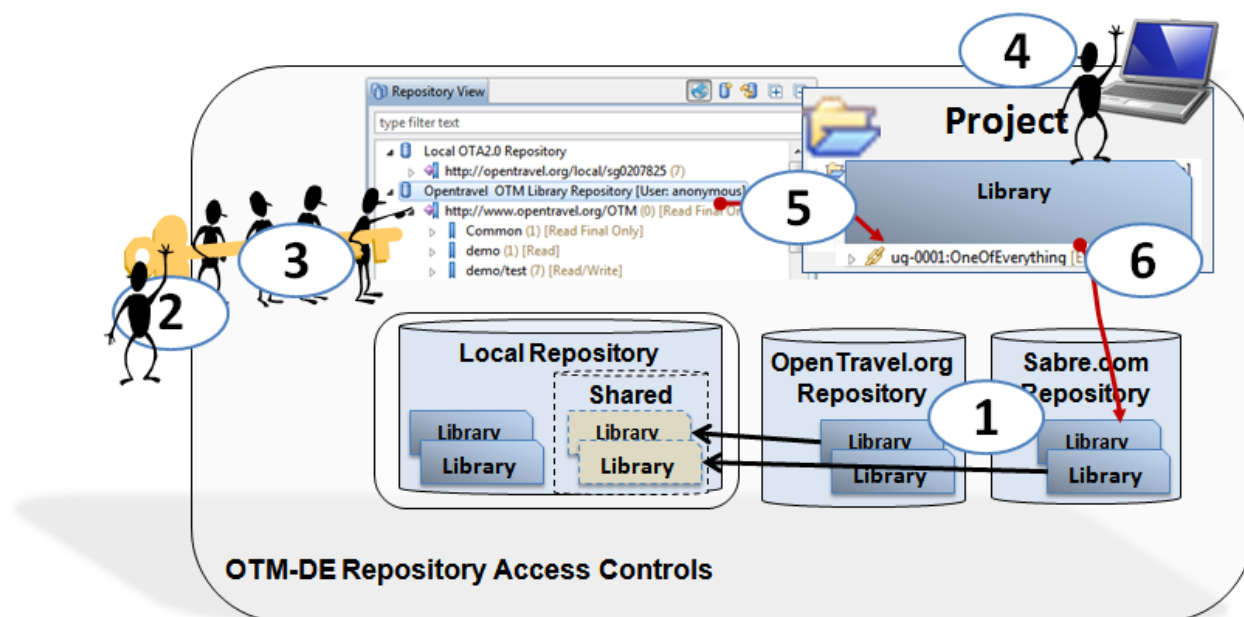
**Storage:** The repository does not directly store libraries; rather it relies upon the system on which it is installed to provide storage. Version 3.1 of the repository supports storage in either local file systems or Subversion (SVN).

## Repository Access Controls

Access Controls in the OTM Library Repository are responsible for assuring that OTM Libraries are shared responsibly and to enforce control over the namespaces used in both the libraries and XSD schemas compiled from the model. OTM Repository access controls are implemented using: users, groups, namespaces, extended-namespaces, libraries and projects.



An overview of the access control process is described here and the details and specific tasks are described in the following sections.

There are three configuration steps:



1. **Repositories** are assigned root-namespaces.
2. **Users** are assigned to groups.
3. **Groups** are granted permissions for namespaces (read-final, read-draft, or write).

When editing the designer follows steps three steps.

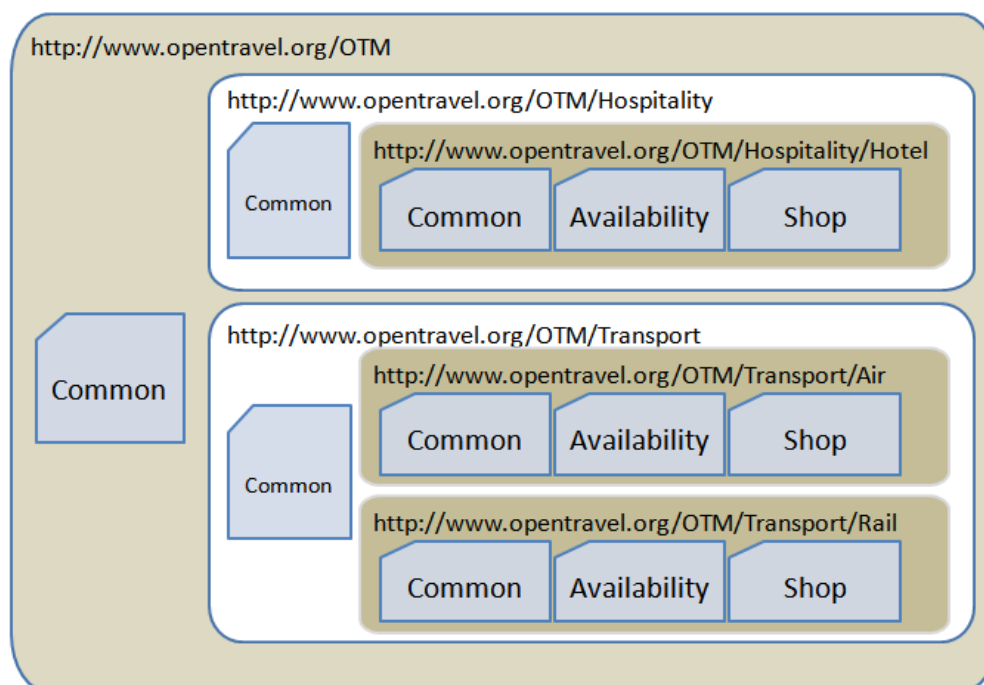
4. **Projects** have a namespace that can be selected from a list of repository managed namespaces or user defined.
5. **Libraries** are *added* to a project. If they are managed in a repository and within the project's namespace they can be *locked* to become *editable*. Locking prevents others from editing the library at the same time.
6. **Unmanaged** libraries (  v2:v2 ) can be *managed* in a repository.  
**Managed**, editable library chains (  ug-0001:OneOfEverything [Editable] [0.1.0] ) can be *committed* back into the repository.

Before a repository will manage a library or allow a library to be committed it will ensure:



1. The library is within the project namespace.
2. The project namespace conforms to the namespaces managed in that repository.
3. The user belongs to a group that has write permission for that namespace.

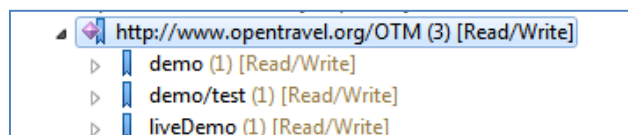
### Namespace Structure

The repository namespaces are designed to be hierarchical. These hierarchies are designed to permit assignment of namespace permissions to teams and to provide a logical namespace for common, shared content.



The above diagram shows 12 libraries in three different managed namespaces. The repository has one or more *Root Namespace(s)* (<http://www.opentravel.org/OTM>) assigned by the repository administrator. The administrator divides the root namespace into *managed namespace(s)* (<http://www.opentravel.org/OTM/Hospitality> and <http://www.opentravel.org/OTM/Travel>). Library designers can sub-divide the managed namespace by providing an *extension* (*Hotel, Air, Rail*) to their project or library.

In this screen capture,  the <http://www.opentravel.org/OTM> root namespace is divided into  three managed namespaces.



## Projects and Libraries

Projects and libraries that use a repository managed namespace as the root of their namespace are eligible to be managed in that repository.

Project and Library menus as well as the Properties view provide drop-down menus to make it easier to select a managed namespace.

Managed Root	<a href="http://www.opentravel.org/OTM">http://www.opentravel.org/OTM</a>
Extension	<a href="http://www.opentravel.org/OTM">http://www.opentravel.org/OTM</a> <a href="http://opentravel.org/local/sg0207825">http://opentravel.org/local/sg0207825</a>

## Users and Groups

The repository administration interface allows the administrator to create groups and add specific users to those groups. Users can belong to multiple groups. When accessing a library in a namespace, the group that the user belongs to with the most access rights will be used.

Group: SchemaAuthors

Use the controls below to add or remove members for this group.

Select an existing user:

- SG0216876
- SG0620641
- SG0899384
- sg0090998
- sg0149852
- sg0191834
- sg0205321
- sg0206003
- sg0207825
- sg0208636

new user:

Add Member ->

<- Remove Member

Selected members:

- sg0207825
- sg0213858
- sg0217473

Save Group Membership Cancel

## Permissions

Repository administrators can grant or deny Read-Final, Read-All and Write for each group to each namespace (root, managed and extended).

opentravel setting the standard OTA2.0 Repository

### Manage Namespace Permissions

Select a namespace from the list below to view the access control settings.

Global Permissions

Namespace: Global Permissions

Group	Grant Permission
Anonymous Users	Read-Final
AWG	Read-Draft
Administrators	Write
SchemaAuthors	Write

## Additional Access Controls


Additional access controls include state management and locking.

**Lifecycle Management:** The repository provides lifecycle management features: the ability to lock and unlock library, versioning and status.

Users can lock a library to prevent editing by other library developers. The holder of the lock (or the administrator) must unlock the library before others can edit it.

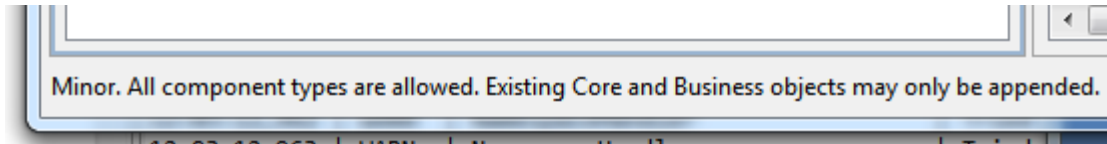
▶ OneOfEverything [Locked: sg0207825] [0.1.0]

Draft/Final status is controlled by the user. When a library is made Final, it cannot be edited or deleted (except by an administrator). A CRC value is added to the library and used to detect changes.

▶  OTM\_Profile\_Demo [Final] [0.0.0]

## Status Bar

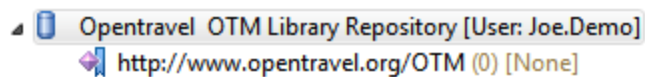
The status bar at the bottom of the Library *Designer* interface gives the user detail about the status and capabilities enabled for a library.



## Task 1 — Proxy Configuration

Most companies use a proxy server to provide security and improve performance and costs associated with accessing the internet. The OTM-DE Model Designer user interface can be configured to use a proxy when accessing remote repositories.

If you are not sure if you need to configure your proxy server settings, try [0 Task 2 — Access a Repository](#), if your result looks like this with “None” as the access permissions and no contents listed then your proxy needs to be configured.



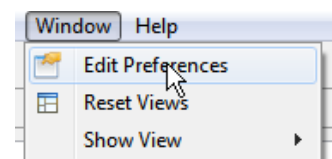
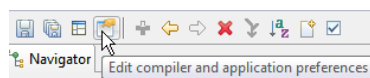
To complete this task you will need:

- The URL for your proxy server including its port number
- User Name and Password if required for your proxy.

Proxy configuration can become a complex task. If these steps do not provide access to remote repositories, please contact your system administrator. For more details on the configuration, see <http://help.eclipse.org/indigo/index.jsp?topic=%2Forg.eclipse.platform.doc.user%2Freference%2Fref-net-preferences.htm>

### 1. Access Preferences View

The view can be opened from the “Window” menu or the Preferences button.



2. Expand the “General” preference and select **Network Connections**

3. Select **Manual** for Active Provider

4. Select **HTTP** and **Edit**

5. Enter your proxy URL

6. Enter Port Number

7. Enter username and password

8. Add a proxy bypass

Enter the your company’s domain name (e.g. \*.your-company.com)

9. If you have already tried to access a remote repository, Refresh the remote repository in repository view.

10. Enter a password.

The Designer keeps your proxy configuration including user name and password in an encrypted location on your system. The password that you enter now will be used whenever OTM-DE Model Designer starts to decrypt the proxy configuration data.

**NOTE** – you will have to enter this password every time you start OTM-DE Designer.

You should now have access to the remote repository.

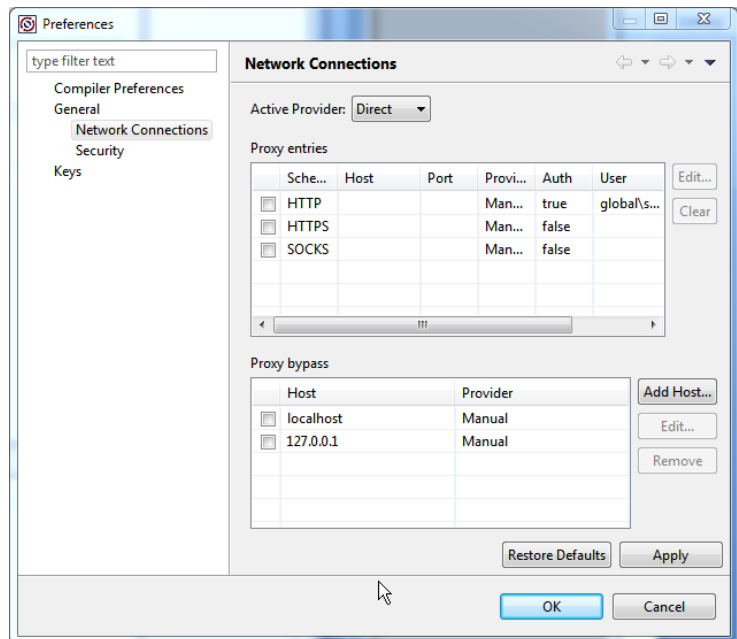
## Task 2 — Access a Repository

In this task you will open a repository. This will only have to be done once, OTM-DE Designer remembers your repositories and opens them each time you start the application.

Before starting this task, you will need:

- The address of a repository
- Optional: An account in that repository
- Optional: Your repository login name and password.
- To have configured your proxy (if needed see task 1).

For an account on the OpenTravel Model Repository please send an email to [otmsupport@opentavel.org](mailto:otmsupport@opentavel.org).





The repository can use an organization's existing domain login accounts to establish user access to an OTM repository. In this case your login and password will be the same you use with other systems in your company.

**1. Select New Repository Button.**

This button is at top of the view. When selected the New Repository wizard will be displayed.

**2. Enter URL: <http://opentravelmodel.net>**

Use the URL provided by the repository administrator.

**3. If you don't have an account, select *Anonymous***

Selecting *finish* will give you access to namespaces that have granted anonymous access.

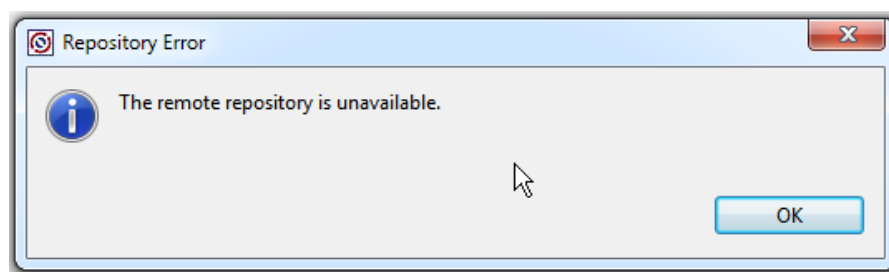
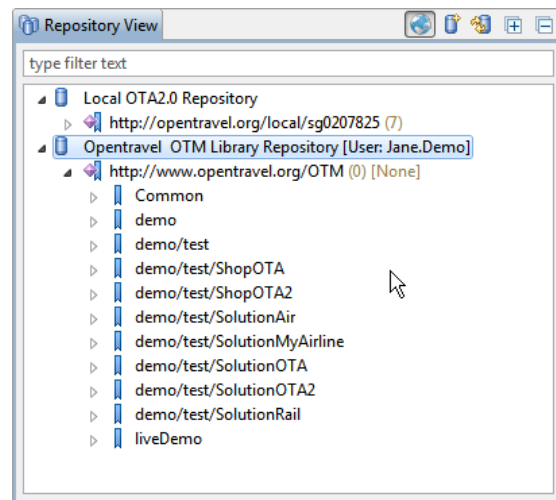
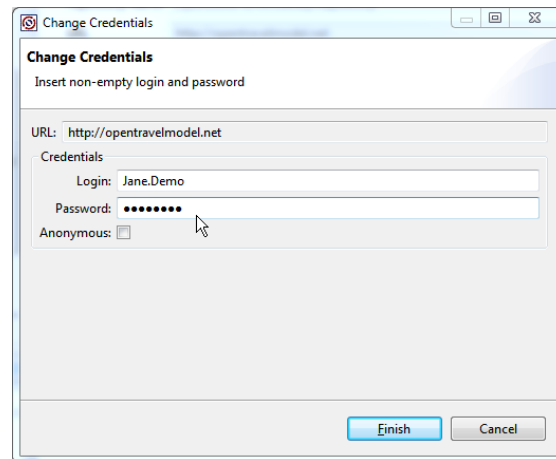
**4. Enter your Login name.**

This name assigned by the administrator.

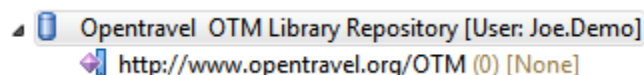
**5. Enter your password.**

**6. Select Finish.**

If you have the repository URL incorrect, you will get an error dialog:



If the repository is opened but there are no libraries and your access rights are "None" then you need to configure your proxy settings (see Task 1 Task 1 — Proxy Configuration).



## Summary

You should now see the repository and its namespaces displayed in the Repository View as shown above. With this view you can browse, search and add libraries to projects.

## Task 3 — Search

The repository can be browsed or searched. There are two search functions: 1) search namespaces and library names and 2) full-text.

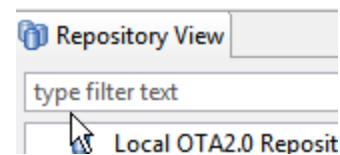
This task assumes that OTM-DE is running and you have 1 or more repositories.

### 1. Select Search Mode

Press the button in to search all of the contents, leave it out to only search file names and namespaces.

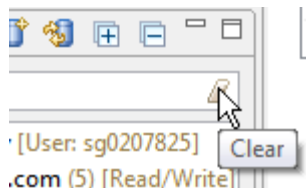
### 2. Enter Search Terms

Enter the terms you are searching for. The repository will only display libraries that contain those terms.



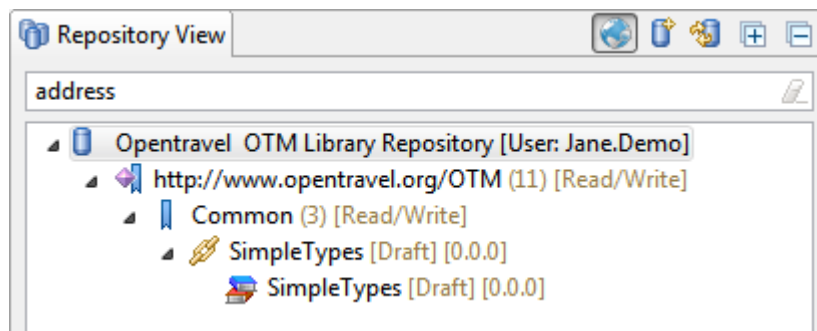
### 3. Clear the search

Press the eraser icon at the end of the search term field to clear the search terms.



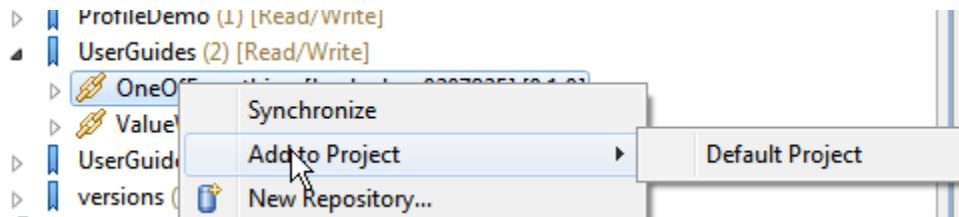
## Summary

The repository search allows you to quickly find libraries that contain words or phrases. The list displayed is filtered to contain only libraries that match the search criterion.



## Task 4 — Accessing Libraries

Libraries are accessed by *adding* the library from the repository to an open project. The repository containing the library must be open (see Task 2 — Access a Repository).

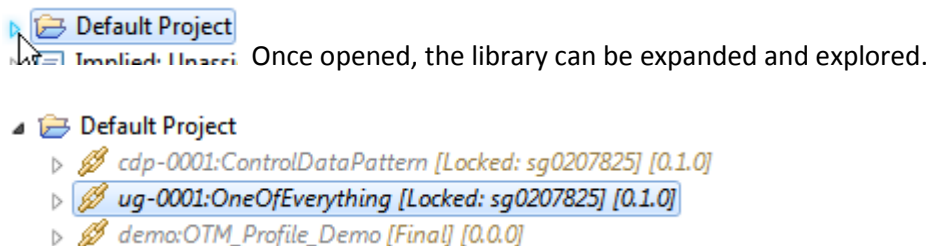


1. Right-click on library in Repository view.
2. Select the project in the cascading menu.

If you are simply reading the library or copying an object out of the repository then you can use any project including the *default* project. The default project is always open and available.

3. Select the project in Navigator view.

If the selected project is not already open, use the little triangle to expand it.



Once opened, the library can be expanded and explored.

## Task 5 — Edit Library

Editing a library requires the library to belong to the project's namespace as described above in the Introduction to [Access Controls](#) section.

1. Find the Namespace and Library in the Repository View
2. If this is a new project then Right-Click and select *New Project*.

Complete the new project wizard. Do **not** change the namespace.

3. If this is an existing project, open the project.

Make sure the project's *Managed Root* encompasses the namespace of the library.

▼ Namespace	
Policy	Namespace policy enforced.
Namespace	<input type="text" value="http://www.opentravel.org/OTM/GettingStarted"/>
Managed Root	<input type="text" value="http://www.opentravel.org/OTM"/>
Extension	<input type="text" value="GettingStarted"/>

4. Right-click on Library and select the project.

5. Right-click on Library and select *Lock*

To edit a library it must be locked. Locking prevents others from editing that library at the same time.

6. Edit

7. UNLOCK

You will hold the lock until you unlock it.