

# **Tagger Eclipse Plugin**

## **Version 1.0**

**Christopher J. Stehno**  
*chris@stehno.com*

Document version 1.0 (11/12/06)

## Table of Contents

1. Introduction.....	3
2. Model Objects.....	3
1. Tag.....	3
2. ITaggable.....	3
3. Data Storage.....	3
1. Tags.....	3
2. Tag Associations.....	4
4. Views.....	4
1. Add New Tag Action.....	4
2. Edit Tag Action.....	4
3. Inline Editing.....	4
4. Remove Tag.....	4
5. Create Working Set From Tags Action.....	4
5. Context Menu Additions.....	4
1. Add Tag Action.....	5
2. Clear Tags Action.....	5
3. Add New Tag Action.....	5
4. View Tags Action.....	5
6. Search Page.....	5
7. View Filters.....	5
8. Wizards.....	5
1. Export Wizard.....	5
2. Import Wizard.....	6
9. Help / Documentation.....	6

## 1. Introduction

The Eclipse Tagger Plug-in will add resource tagging functionality to Eclipse similar to that provided by the Del.icio.us bookmark management website or the GMail webmail client. It will provide a means for managing, finding and filtering tags and tagged resources.

This tagging is desirable such that it would be useful to be able to group files in simple, cross-project manner based on arbitrary criteria. This functionality will be integrated into the working set functionality to allow the creation of working sets based on tagged resources thereby adding value to this pre-existing feature as well as adding new functionality.

## 2. Model Objects

There will be one required model object and an interface that will be used to adapt IResource elements for use by the Tagger plugin.

### 1. Tag

The Tag object will have the following properties:

long id	The unique identifier.
String name	The name of the tag.
String description	A short description of what the tag is used for.

### 2. ITaggable

An IAdapterFactory will be used to adapt IResource implementations to the ITaggable interface so that tags associated with the object may be managed. This interface will have the following methods:

void setTag(long id)	Sets (adds) the tag with the specified id.
void clearTag(long id)	Clears (removes) the tag with the specified id.
void clearTags()	Clears (removes) all tags from the item.
long[] listTags()	Lists all tags (ids) associated with the item.
boolean hasTag(long id)	Returns true if the specified tag is associated with the item.

## 3. Data Storage

Data storage will be very lightweight and simple, utilizing pre-existing storage and

serialization functionality native to Eclipse.

## **1. Tags**

The set of available tags will be managed by the plugin and will be stored with the plugin state using the standard Eclipse state persistence mechanism.

## **2. Tag Associations**

The tag association data will be stored as a persisted attribute of each tagged IResource object. Only the tag ids will be persisted to minimize data storage and lessen the potential for tag data versioning issues (e.g. stored tag information does not match what is contained in the plugin state data).

## **4. Views**

A Tags View will be added so that the tag set can be managed by the user. The view will have tool bar actions that allow the management of the available tag set.

### **1. Add New Tag Action**

Used to create a new tag to add to the tag set. This action will pop-up a simple dialog so that the user can enter a title and description for the tag.

### **2. Edit Tag Action**

Used to edit the properties of a tag. This action will pop-up a simple dialog (the same as used in the [Add New Tag Action](#)) to allow the user the ability to change the name and description of the selected tag.

### **3. Inline Editing**

Double clicking on a tag entry will enable in-line editing of the name and description for the tag.

### **4. Remove Tag**

Used to remove the selected tag from the tag set. This will invalidate any resource associations with the tag. The resource should be able to gracefully handle this loss of the associated tag. A confirmation alert will be given to ensure that the deletion is really desired.

### **5. Create Working Set From Tags Action**

Used to create a working set from all resources that are associated with the specified tags. A simple dialog will be presented to the user so that the name of the working set may be entered.

## 5. Context Menu Additions

A “Tags” submenu will be added to the context menu (pop-up) of IResource implementation.

### **1. Add Tag Action**

Used to add an existing tag to the selected resources tag set. The user will be shown a simple dialog with a list of all available tags. Tags already associated with the resource will not be available. If more than one resource is selected, the dialog will not show those tags common to all selected resources, but will show those associated with some of the selected resources (they will be shown differently but will be selectable).

### **2. Clear Tags Action**

Used to clear all tags from the selected resource(s). A confirmation alert will be presented to ensure that this is a desired action.

### **3. Add New Tag Action**

Used to create a new tag and add it to the selected resource(s). This action will present the user with the same dialog used by the [Add New Tag Action](#).

### **4. View Tags Action**

Used to view the tags associated with the selected resource(s). The dialog displayed will be the same one as used by the [Add Tag Action](#).

## 6. Search Page

A search page will be added to provide a means for finding resources based on their associated tag or tags. This search page may also provide other simple search criteria such as by file extension or text content.

## 7. View Filters

Package Explorer and Project Navigator view filters will be provided so that these views may be filtered to show only those resources with the selected tag(s).

## 8. Wizards

Two wizards will be added: One for exporting the tag set and one for importing exported tag sets.

### **1. Export Wizard**

An export wizard will be provided to allow the export of the tag set for the plugin.

This export should be in a simple xml-based format. The wizard page will allow the user to specify the file to be exported into.

## ***2. Import Wizard***

An import wizard will be provided to allow the import of tag sets from exported tag set data. The wizard page will allow the user to select the file to be imported.

## **9. Help / Documentation**

Usage and contextual help documentation will be developed including “how-to”s with screen shots. The should be provided using the standard Eclipse help/documentation system.