

# Intrallect intraLibrary 3.1: Administrator's Guide

Revision: 7

Created: 4th August 2005

Last Revised: 30th June 2009

Contact: [support@intrallect.com](mailto:support@intrallect.com)

Company: [Intrallect Ltd](#)

Product: [intraLibrary](#), [Digital Repository](#)

Copyright: Intrallect Ltd 2003-2009. All rights reserved.

**This document is made available to support Intrallect's customers and users of Intrallect's software. The text of these documents and the design of the intraLibrary software are both the intellectual property of Intrallect Ltd. Intrallect does not provide this document for any other purpose, and offer no warranty nor accept any liability for its use in any other context.**

---

## Table Of Contents

---

|   |    |
|---|----|
| 1. Introduction.....                                      | 6  |
| 2. Users .....  | 6  |
| 2.1. User Administration.....                             | 6  |
| 2.2. System Settings for Users .....                      | 7  |
| 3. Groups and Workflows .....                             | 7  |
| 3.1. Introduction .....                                   | 7  |
| 3.2. Submission Workflow .....                            | 9  |
| 3.2.1. Properties of a Workflow .....                     | 9  |
| 3.2.2. Properties of a Workflow Stage.....                | 10 |
| 3.2.3. Workflow Actions.....                              | 11 |
| 3.2.4. Process Completion Criteria.....                   | 13 |
| 3.2.5. Example Workflow Definitions .....                 | 15 |
| 3.2.5.1. Example 1: Basic intraLibrary Workflow .....     | 15 |
| 3.2.5.2. Example 2: Contribute, Catalogue, Review .....   | 15 |
| 3.2.5.3. Example 3: Advanced Cataloguing Workflow.....    | 16 |
| 3.2.5.4. Example 4: Simple Content Managing Workflow..... | 16 |
| 3.2.6. Working on Resources .....                         | 17 |
| 3.2.7. Defining a Workflow for Use in intraLibrary .....  | 17 |
| 3.2.8. Removing Users and Groups .....                    | 18 |
| 3.3. Working in Groups and Workflows .....                | 18 |
| 3.3.1. Relationship Between Groups and Workflows .....    | 18 |
| 3.3.2. Access to Resources in a Workflow .....            | 18 |
| 3.3.3. Work Area .....                                    | 19 |
| 4. Collections .....                                      | 20 |
| 4.1. Introduction .....                                   | 20 |
| 4.2. Properties of Collections .....                      | 20 |
| 4.2.1. Summary of Collection Properties.....              | 20 |
| 4.2.2. Relationship to Classification Systems.....        | 22 |

|  |    |
|--|----|
| 4.2.3. Relationship to Application Profiles and Licence Models ..... | 22 |
| 4.3. Managing Collections.....                                       | 22 |
| 4.4. Managing Access .....   | 23 |
| 4.4.1. Relationship to Groups of Users .....                         | 23 |
| 4.4.2. Access Permissions .....                                      | 23 |
| 4.4.3. Contributing to a Collection .....                            | 24 |
| 4.5. Controlling Remote Access to Collections .....                  | 24 |
| 4.6. An Example of Using Collections.....                            | 25 |
| 4.7. Precedence Issues .....   | 26 |
| 5. Metadata and Vocabularies .....                                   | 26 |
| 5.1. Introduction .....  | 27 |
| 5.2. Configure Application Profiles.....                             | 27 |
| 5.3. Edit an Application Profile .....                               | 28 |
| 5.4. Manage Vocabularies .....                                       | 29 |
| 5.5. Metadata Subsets .....  | 30 |
| 6. Search .....  | 30 |
| 6.1. Configure Search .....  | 30 |
| 6.1.1. Simple Search.....  | 30 |
| 6.1.2. Advanced Search.....  | 31 |
| 6.1.3. Similarity Search .....                                       | 31 |
| 6.1.4. Reference Search .....  | 31 |
| 6.2. Public Saved Searches.....                                      | 32 |
| 7. Agreements .....  | 32 |
| 8. Licences .....  | 32 |
| 8.1. Introduction .....  | 32 |
| 8.2. Licensing Model.....  | 33 |
| 8.3. Depositing Resources.....                                       | 33 |
| 8.4. Accessing Resources .....                                       | 34 |
| 8.5. Configuring the System.....                                     | 34 |
| 8.6. What Community Licensing Doesn't Do .....                       | 35 |

|  |    |
|--|----|
| 8.7. Technical Solution.....                 | 35 |
| 8.8. References on Licensing .....           | 35 |
| 9. System .....                              | 36 |
| 9.1. General Configuration.....              | 36 |
| 9.1.1. User Options .....                    | 36 |
| 9.1.2. Metadata Options .....                | 36 |
| 9.1.3. RSS Options.....                      | 37 |
| 9.1.4. System Options .....                  | 37 |
| 10. Support Login .....                      | 37 |
| 10.1. When You Need Support.....             | 37 |
| 10.2. What Support Access Is.....            | 38 |
| 10.3. Activating Support Access .....        | 38 |
| 10.4. De-Activating Support Access .....     | 38 |
| 11. Mail Configuration .....                 | 39 |
| 11.1. Email Basics .....                     | 39 |
| 11.2. Configuring the SMTP Settings.....     | 39 |
| 11.3. Testing Mail Settings.....             | 39 |
| 11.4. Publication Notification.....          | 40 |
| 12. Metadata Reports .....                   | 40 |
| 12.1. Configuring Metadata Reports .....     | 40 |
| 12.2. Saving Metadata Report Templates ..... | 41 |
| 12.3. Repository Statistics Table .....      | 41 |
| 13. Maintenance .....                        | 42 |
| 13.1. Resource Cache .....                   | 42 |
| 13.2. Search Index (Re-index Metadata) ..... | 42 |
| 13.3. Image Thumbnails .....                 | 42 |
| 13.4. Metadata Templates .....               | 43 |
| 14. Metadata Templates .....                 | 43 |
| 14.1. Introducing Metadata Templates .....   | 43 |
| 14.2. Creating Metadata Templates.....       | 43 |

14.3. Service Result Templates ..... 44

14.4. Template Format..... 44

15. Email Templates ..... 46

15.1. Email Alerts for Stage Change..... 46

15.2. Email Alerts for Publication of a Resource..... 46

15.3. Email Alerts for Annotation of a Resource ..... 47

15.4. Configuring the Email Templates..... 47

15.5. Email Template Macros ..... 48

16. Appendix 1: Metadata Template Macros..... 49

## 1. Introduction

This document is for intraLibrary administrators carrying out routine administration tasks. For support with decision-making about configuration tasks see the **Configuration Guide**.

One or more users may have an administration role in an installation of intraLibrary. When they log-in to intraLibrary these users see an extra *admin* tab at the top of the screen.

When the *admin* tab is selected an additional toolbar becomes available, additional tab options become available:

- users;
- groups;
- workflows;
- collections;
- metadata;
- vocabularies;
- search;
- agreements;
- licences;
- system;
- support;
- mail;
- reports.

This document describes the facilities that are available in the *admin* tab, and the issues that must be considered before changing some of the settings.

## 2. Users

### 2.1. User Administration

The "users" section provides facilities to add, delete and update users.

Users can be created in the following roles:

**Guest:**

can search and browse the repository, and export resources.

**Contributor:**

can additionally upload, catalogue and publish resources.

**Librarian:**

can additionally delete, move and unpublish published resources, and edit the classification systems.

**Administrator:**

can manage user accounts and configure the system.

Your local licence for your intraLibrary installation constrains the number of users that can be added in the contributor, librarian and administrator roles.

## 2.2. System Settings for Users

In the [system "user options" area](#) the administrator can choose a number of general configuration options that will affect users.

# 3. Groups and Workflows

## 3.1. Introduction

It is hard to separate the functions of groups from workflows as the two are tightly linked, so they will both be treated together. First the concepts of the submission workflow are described then how to configure a workflow including setting up particular groups of users with particular roles. It should be stressed that intraLibrary has a default workflow which many people will find satisfies their needs and if that is the case there is no need to read this section.

Several new terms will be introduced in this section so it is useful to start with some definitions:

**submission workflow**

The process of uploading a resource into intraLibrary and preparing it to ensure that all necessary information is provided and checked.

**stages**

The submission workflow can be divided into several stages (typically one to four

stages). Stages occur sequentially in a workflow.

### **completion criteria**

A resource is unable to move from one stage to the next until it has satisfied all the completion criteria for that stage.

### **stage transitions**

When a resource moves from one stage to the next a stage transition occurs. This transition can be used to trigger events.

### **group member**

Users may be assigned to one or more groups and a submission workflow can be assigned to a group. A user can be a member of several groups. One workflow can be assigned for use by several groups.

### **group roles**

A group member can be given a specific role in that group. Roles can be named by the administrator. A user can have different roles in different groups. Group roles are used to give users access to workflow processes.

### **group alert**

When a resource undertakes a stage transition a group alert may be sent. A group alert takes the form of an email sent to all users involved in the next stage.

### **workflow processes**

Within a single stage of a submission workflow there can be one or more processes. If more than one process exists then these processes can be carried out in parallel. Processes are defined by grouping together a number of workflow actions which can be used to complete that process.

### **workflow actions**

Workflow actions are predefined (such as edit metadata, or preview resource) and are associated with a workflow process.

### **upload**

The action of submitting a resource into intraLibrary. Uploading is carried out in the section of the work area called upload.

### **available**

Resources are available, and can be seen in the available section of the work area, when a user can work on them. Only resources at a stage of a workflow for which the user has been assigned a role will be visible. Users can elect to work on any resources in the available area by reserving them.

### **reserved**

When a user reserves a resource it appears in the reserved section of that user's work area and becomes unavailable to other users, though they can see that the resource



exists and who has reserved it. If the user is the only person with access to a particular stage in a workflow, resources in that stage will automatically go to the reserved area.

### 3.2. Submission Workflow

A submission workflow defines the steps that a resource passes through as it is uploaded, catalogued and published into the repository. IntraLibrary has a highly configurable workflow. It enables complex, multi-role workflows to be defined, whilst still supporting the simplicity of a single-stage workflow where appropriate.

#### 3.2.1. Properties of a Workflow

The progress of a resource through its submission to the repository is defined by the stages of the submission workflow. When a resource progresses through a workflow, stage transitions can trigger certain events, such as a change in the resource's published state and sending alerts to users with relevant workflow roles. When resources are moved in a workflow, they are moved to a particular stage, such as being sent back to a previous stage for clarification, or progressing to the next stage for classification.

Each stage of a submission workflow can have one or more workflow processes assigned to it. This means parallel processes can be defined, so users in different roles can potentially access the resource at the same stage, and do different things to it. The things that can be done to a resource are known as workflow actions. The list of available actions is given below.

Group members can be assigned one or more roles in a submission workflow. A role can be associated with one or more processes in the submission workflow. This flexible arrangement means it is possible to create a variety of different ways for people to collaborate in a submission process. It anticipates the need for users in the system to contribute to the submission of resources in more than one way, and the need for special users who have access to most or all of the workflow.

To summarise the properties of a submission workflow:

- A workflow consists of several stages.
- Each stage can have one or more processes that can be accessed in parallel.
- Any group role can be given access to one or more workflow processes (multiple group

- roles can be defined for a group of users).
- Each workflow process can include one or more predefined actions which are made available to users in the relevant role(s).
- Some workflow actions can be configured to require completion within a workflow stage before the resource can be moved to the next stage.

### 3.2.2. Properties of a Workflow Stage

As described above, it is a property of every workflow stage, that all users who have access to that stage will receive email alerts when resources arrive in that stage (although the users can configure this themselves in their "profile"). A workflow stage can have the following additional properties.

| Property             | Description  | Additional Parameters |
|----------------------|--|-----------------------|
| Change publish state | The publish state of the resource will change when the resource has passed through the stage. By default, resources are not published when they enter the first stage of a workflow, even if they were published before they entered the workflow. Setting this property at any stage, including the first stage, causes the resource to be published so it can be seen by other users. Each further use of this property toggles the published state of the resource. |                       |
| Complete on any      | Where a workflow stage has more than one process, by default all processes must be completed in order to move the resource to the next stage. Setting this property means that the resource may be moved to the next stage by completing only one process in the stage.  |                       |

|                          |  |                 |
|--------------------------|--|-----------------|
| Alert resource owner     | Resource owner will be alerted when stage is completed. This is to allow the contributor of a resource to be alerted when something happens to their resource, for example when it is withdrawn from the repository. |                 |
|                          | Alert message configurable by modifying the email template "stageChangeObjectOwnerEmail.txt". See <a href="#">Email Templates</a> section for more information.  |                 |
| Email addresses to alert | Emails will be sent to the supplied addresses when a resource has passed through the stage. This feature is designed for sending alerts to external entities, such as mailing lists.                                 | Email addresses |
|                          | Alert message configurable by modifying the email template "stageChangeExternalUserEmail.txt". See <a href="#">Email Templates</a> section for more information.   |                 |

### 3.2.3. Workflow Actions

There is a set of predefined actions that are available to be included in any workflow process. Some actions may have [completion criteria](#) such as conformance to a subset of an application profile, or matching one of a set of licence patterns.

| Action             | Workflow XSD Element | Description  | Additional Parameters                         |
|--------------------|----------------------|--|---|
| Change collections | change_collections   | Modify which collections the resource is a member of. User can put the resource in any collections they have contribute access to. |   |
| Delete             | delete               | Remove a resource completely from the system.  |   |
| Edit metadata      | edit_metadata        | Metadata / classification editor   | Metadata Subset. See <a href="#">Metadata</a> |

|                        |                         |   |                                  |
|------------------------|-------------------------|---|----------------------------------|
|                        |                         | configured by the chosen Application Profile, and an optional metadata subset.  | <a href="#">Subsets</a> section. |
| Edit rights            | edit_rights             | Licence (rights management) editor.   |                                  |
| Export                 | export                  | Package export functions.   |                                  |
| Import from filesystem | import_from_file_system | Import a file or content package from the group's folder on the server filesystem.  |                                  |
| Link resources         | link_related_objects    | Link a resource to any other resource by populating the LOM Relation fields for both resources. All resources must have the correct LOM field and vocabulary assigned in the relevant application profile, and must have title and LOM identifier fields populated. |                                  |
| Move resource          | move_object             | Send the resource to another stage of the workflow, with comments attached. Typically used at review stages.  | Stage ID                         |
| Overwrite              | overwrite_resource      | Upload a new version of a resource to the system. New version has same metadata as old resource. The  |                                  |

|                             |                              |  |                                      |
|-----------------------------|------------------------------|--|--------------------------------------|
|                             |                              | exception is if a new title has been entered at upload or imported in a content package.   |                                      |
| Preview                     | preview                      | Package/file preview.  |                                      |
| Return                      | return                       | Puts the resource back into the workflow and published state it had before it entered the current workflow.                              |                                      |
| Save manifest to filesystem | save_manifest_to_file_system | Write a package manifest to a folder on the server filesystem. Folder location is a property of the workflow process.                    | Path of folder on server filesystem. |
| Save to filesystem          | save_to_file_system          | Write a content package to a folder on the server filesystem.  | Path of folder on server filesystem. |
| Transfer                    | transfer                     | Move the resource into a different workflow stage or different group. User can move the resource to any group/stage they have access to. |                                      |
| Upload                      | upload                       | Upload or add a new resource to the system.  |                                      |
| View Metadata               | view_metadata                | Metadata preview.  |                                      |

### 3.2.4. Process Completion Criteria

As mentioned above, some workflow actions may have criteria that need to be met before the process can be completed. In the user interface, the process completion button will be greyed out if these criteria have not been met. The user can still click on the button, but instead of marking the process as complete, it will display details of the completion criteria that have not been satisfied.

Process completion criteria are associated with specific actions. There are currently four actions which have process completion criteria:

- Edit Metadata
- Edit License
- Save Package to Filesystem
- Save Manifest to Filesystem

The following paragraphs give details of the process completion criteria for each of these actions.

The process completion criteria for the "Edit Metadata" action are driven by the [application profile](#). This can be the whole application profile, or a part of the application profile defined in a metadata subset for a workflow process. If the "Edit Metadata" action is included in a workflow process, the metadata record must meet the conditions of the application profile and any metadata subset, before the workflow process can be completed.

The conditions that can be imposed by an application profile include:

- Completion of all mandatory fields, such as "Title", "Description", "Contributor". NB: If a metadata subset has been defined for the workflow process, only the mandatory fields that are included in the metadata subset must be completed to meet the completion criteria for that process.
- Use of the vocabularies required for certain fields.

The process completion criteria for the "Edit Licence" action are driven by the [licence model](#). If the "Edit Licence" action is included in a workflow process, the licence information must match one of the licence patterns in the licence model before the workflow process can be completed.

The conditions that can be imposed by a licence pattern include:

- Matching the fixed properties of a particular licence.
- The completion of certain fields, such as "Resource Owner".
- Particular values being stored in certain fields.

The "Save Package to Filesystem" and "Save Manifest to Filesystem" actions both have only one process completion criterion. The user must have performed the action before the process can be completed. In technical terms, these two actions have a "state", i.e. the system remembers whether or not the user has performed that action.

If a workflow stage has more than one process defined, and more than one of these processes has completion criteria, then all of the completion criteria for all processes in a stage must be met before a resource in that workflow can be moved to the next stage. The exception is if the workflow has the property "Complete On Any" set, which means that only one process needs to be completed to move the resource to the next stage.

### 3.2.5. Example Workflow Definitions

Intrallect can create workflows to customer's requirements, or you can [define your own workflows](#) and import them into intraLibrary. The following workflows are given as examples of what is possible. Each defines one or more processes, related actions and the published state (published/unpublished) in a workflow with one or more stages.

#### 3.2.5.1. Example 1: Basic intraLibrary Workflow

|               |   |
|---------------|---|
| <b>Stages</b> | <b>Stage1: Contribute</b>   |
| Processes     | Process 1: Upload and Catalogue <ul style="list-style-type: none"> <li>• Upload</li> <li>• Edit Metadata (All)</li> </ul> |
| Publish State | Unpublished   |

#### 3.2.5.2. Example 2: Contribute, Catalogue, Review

| <b>Stages</b> | <b>Stage1: Contribute</b>  | <b>Stage2: Catalogue</b>  | <b>Stage3: Review</b>   |
|---------------|--|---|---|
| Processes     | Process 1: Upload <ul style="list-style-type: none"> <li>• Upload</li> <li>• Preview</li> <li>• Edit Metadata <ul style="list-style-type: none"> <li>• ("General" subset)</li> </ul> </li> </ul> | Process 1: Catalogue <ul style="list-style-type: none"> <li>• Preview</li> <li>• Edit Metadata (All)</li> </ul> | Process 1: Check <ul style="list-style-type: none"> <li>• Preview</li> <li>• Preview metadata</li> <li>• Reject (to stage 2)</li> </ul> |

|               |             |             |           |
|---------------|-------------|-------------|-----------|
| Publish State | Unpublished | Unpublished | Published |
|---------------|-------------|-------------|-----------|

### 3.2.5.3. Example 3: Advanced Cataloguing Workflow

| Stages        | Stage1: Contribute  | Stage2: Clear Rights   | Stage3: Catalogue   | Stage4: Classify  |
|---------------|---|--|---|---|
| Processes     | Process 1: Upload <ul style="list-style-type: none"> <li>• Upload</li> <li>• Preview</li> <li>• Edit Metadata ("General" subset)</li> </ul> | Process 1: Rights <ul style="list-style-type: none"> <li>• Preview</li> <li>• Edit Rights</li> </ul> | Process 1: Transcription <ul style="list-style-type: none"> <li>• Preview</li> <li>• Edit Metadata (General and Technical)</li> </ul> Process 2: Subject <ul style="list-style-type: none"> <li>• Preview</li> <li>• Edit Metadata (Subject)</li> </ul> Process 3: Education <ul style="list-style-type: none"> <li>• Preview</li> <li>• Edit Metadata (Education)</li> </ul> | Process 1: Classify <ul style="list-style-type: none"> <li>• Preview</li> <li>• Edit Metadata (Classification)</li> </ul> |
| Publish State | Unpublished   | Published  | Published   | Published   |

### 3.2.5.4. Example 4: Simple Content Managing Workflow

| Stages    | Stage1: Create Placeholder  | Stage2: Add Content  | Stage3: Review Content  |
|-----------|---|--|---|
| Processes | Process 1: Upload <ul style="list-style-type: none"> <li>• Upload</li> <li>• Edit Metadata (All)</li> </ul> | Process 1: Upload <ul style="list-style-type: none"> <li>• Preview</li> <li>• Overwrite</li> <li>• Edit Metadata (Technical subset)</li> </ul> | Process 1: Check <ul style="list-style-type: none"> <li>• Preview</li> <li>• Move (to stage 2)</li> </ul> |



|               |             |             |           |
|---------------|-------------|-------------|-----------|
| Publish State | Unpublished | Unpublished | Published |
|---------------|-------------|-------------|-----------|

### 3.2.6. Working on Resources

Users must reserve a resource before they can work on it. This can happen in one of three ways:

- The user is the person who uploaded the resource, and they ensured that the "reserve" checkbox on the upload page was checked.
- The user is the only person with access to the stage the resource is in. In this case the resource automatically goes into their reserved area.
- The user clicks on the "reserve" button for that resource from the available section of their work area.

Once they have reserved the resource, it appears in their reserved area and the actions in the relevant workflow processes become available. The actions can be accessed by selecting from dropdown menus available by clicking the small or large cog icons. They also have a blue arrow icon to click when the process has been completed.

In the reserved area the user can work on the resource in various ways: edit metadata, manage licensing etc. When they have completed their work, and met any criteria on the metadata, they click the button to declare the process as completed.

When all the processes at that stage are marked as completed, the resource will be moved on to the next stage. Alternatively, if that workflow stage has the property "Complete On Any" set, then only one process for that stage needs to be completed for the resource to be moved to the next stage. Users who have access to the resource at the next stage will receive an email alerting them that there is a new resource to work on, unless they have changed their user profile to not receive group alerts.

### 3.2.7. Defining a Workflow for Use in intraLibrary

A workflow definition can be added to intraLibrary by a user who has administrator access to the repository. To do this, you must have created your workflow definition in an XML document which conforms to the XML Schema for submission workflows, which is publicly available at [http://www.intrallect.com/xsd/intralibrary\\_workflow\\_v0p5.xsd](http://www.intrallect.com/xsd/intralibrary_workflow_v0p5.xsd). See the intraLibrary **Configuration Guide** for more details on how to configure a workflow.

### 3.2.8. Removing Users and Groups

An administrator can remove a contributor from the system. If there are resources which the contributor is working on, the administrator will have the option to have the resource's ownership information removed.

Administrators can also remove whole groups from the system. If the group is currently working on resources then the administrator will have the option to transfer the resources to another group.

## 3.3. Working with Groups and Workflows

### 3.3.1. Relationship Between Groups and Workflows

There may be one or more workflows available in the system. After creating a group of users, an administrator can assign one of the available workflows assigned to the group. This group of users shares an instance of the workflow. Other groups can share their own instances of the same workflow.

Users get access to processes within a workflow by being assigned workflow roles within the group. To give roles access to workflow processes, the administrator must add the roles assigned to the process in the workflow details page. The administrator assigns users to one or more workflow roles in the context of a particular group. Users who are members of more than one group can be given different roles in each group.

It's also possible to ensure all users in a group have access to their own resources when those resources are in a particular workflow process. The process must include the element `ownerAccess` within the workflow. The intraLibrary default workflow, for instance, only has one stage for upload and cataloguing, and has `ownerAccess` (and no group roles) assigned, so that all users can upload their own resources, catalogue, and publish them.

### 3.3.2. Access to Resources in a Workflow

To participate in a submission workflow, users must have contributor status. Normal users do

not have a [Work Area](#).

Resources can only be associated to one group, and are therefore part of only one submission workflow. Resources exist at a single stage in the instance of the workflow belonging to that group. Group members assigned a role in a submission workflow may have access to one or more stages of the workflow.

Users share an "available" area for each stage of the group's workflow. All users in the relevant roles will be alerted when a resource is moved into the relevant "available" area. Any user in the relevant roles can then choose to select this resource to work on (see below).

To allow a contributor user to add a resource to a particular collection, the administrator must [give the group "contribute" permission to the collection](#). A resource must be in at least one collection and can be in many different collections.

### 3.3.3. Work Area

Each contributor user has a work area where they participate in the submission workflow. The work area is divided into three parts: Reserved; Available; and My Resources. When the user has group role access to a workflow stage that includes the action "upload", they will have an upload button available. Likewise, if they have access to a workflow stage that includes the action "import from filesystem", they will also have an import button. If they do not have group access-permission to contribute to any collection, neither the upload nor import button will appear in their work area.

In the *available* section users can see all the resources they potentially have access to. Each resource has a listing that includes title and description, and access to basic functions such as preview. Users can select the resources they want to reserve so that they can work on them.

In the *reserved* section you see all the resources you have chosen to work on. In addition to the basic information and functions, each workflow process that is available for the resource is accessible through clicking on the small or large cog icons.

In both the *available* and *reserved* areas, the list of resources displayed can be constrained by various criteria. Users can apply filters to select resources in a particular group and a

particular stage of a workflow. [Saved searches](#) can be used to search the list of resources by specific properties.

The *upload* page, which appears when the user clicks the upload button, is where a user can contribute a resource (file, pages, URL, content package, etc.) to the system. Users only see this section if they have access to a workflow process that includes the ["upload" action](#). Users can choose at the upload stage whether or not to automatically reserve resources they contribute. Users who are members of several groups can choose which group they are contributing the resource to. If the group has access to more than one collection then they can choose which collections to put the resource into. They can also upload any number of resources at once by clicking the '+' sign in the upload form.

The *import* section is provided to enable users to import large files from the server filesystem. Import also allows users to upload a number of resources at once. Users only see this section if they have access to a workflow process that includes the ["import from filesystem" action](#). When users import resources, they can choose whether or not to automatically reserve those resources. Users who are members of several groups can choose which group they are contributing the resources to. If the group has access to more than one collection then they can choose which collections to put the resources into.

## 4. Collections

### 4.1. Introduction

Collections are used in intraLibrary to collect together resources which have a common property, such as ownership or resource type. There are various reasons why a collection might be created, but the main practical reason is for controlling different kinds of access to a set of resources, or what kinds of metadata or licences are applied to them. Different levels of access to a collection can be assigned to different groups of users, for the purpose of controlling which users can see what resources, and which parts of the repository users can contribute to.

### 4.2. Properties of Collections

#### 4.2.1. Summary of Collection Properties

There are a number of properties which can be set for a collection:

- **Name:** A name for the collection.
- **Description:** A description for the collection.
- **Identifier:** A string that can be used to identify the collection.
- **Application Profile:** The [metadata application profile](#) to be applied to resources in the collection (optional).
- **Licence Model:** The [licence model](#) to be applied to resources in the collection (optional).
- **Show Package Navigation:** When resources in IMS or SCORM Content Package format are previewed, or viewed via a Public URL, by default they are displayed with a left-hand frame indicating the structure of the package, allowing the user to navigate through the package, or see a flat list of resources in the package. In addition, when a user has "package" set as their default resource view in their preferences, they see single file resources within the package view left-hand frame. Un-checking the Show Package Navigation option means that packages and single file resources will be displayed in all cases without the left-hand frame, with the first page appearing by default. If a resource is in more than one collection, and the collections have different settings for this option, then the package navigation frame will be shown. Switching off this option is useful if you have packages that have their own navigational structure built into them; users then don't need the left-hand frame. NB: If you switch this setting off, users won't have access to "export package" via Public URL, as noted below under "Allow Public Package Download".
- **Default Permissions:** The [access permissions](#) that are applied to groups without specific permissions (normally the minimum level of access).
- **Access to Taxonomies:** The classification system(s) that can be applied to resources in the collection, and viewed by users with "Find" access to the collection.
- **Searchable by External Systems:** Whether or not external systems can search the collection using SRU/SRW.
  - More information on this is available in the [Controlling Remote Access to Collections](#).
- **Harvestable by External Systems:** Whether or not external systems can harvest metadata about published resources in the collection using OAI-PMH.
- **Allow Public Preview:** Whether or not external users can preview resources in the collection using a Public URL.
- **Allow Public Package Download:** Whether or not external users can download packaged versions of resources in the collection, using a Public URL. NB: The Public Package URL is made available as an "export package" link in the left-hand package navigation frame when viewing the package via Public URL. If the "Show Package Navigation" setting above is switched off, this link will be unavailable regardless of whether Allow Public Package Download is switched on. A Public Package URL can also be generated via a metadata template, which can be applied to the resource's metadata in the metadata editor, or at the point of SRU/SRW searching or OAI-PMH harvesting.

### 4.2.2. Relationship to Classification Systems

One or more classification systems (taxonomies) can be used for each collection. This enables shared classification systems to be used for some or all resources in the repository, or separate classifications to be used for each collection.

The classification systems used for collections affect what users see in their Browse Library interface, and contributing users see in their Classification interface. For example, if collection A uses classification system A and collection B uses classification systems B and C then:-

- A user who only has access to collection A will see classification system A.
- A user who only has access to collection B will see classification systems B and C.
- A user who has access to collections A and B will see classification systems A, B and C.

### 4.2.3. Relationship to Application Profiles and Licence Models

Collections in intraLibrary can use different metadata application profiles and licence models from each other. Many customers may have no need for this feature, so it is important to point out that many installations of intraLibrary will only ever use one metadata application profile and one licence model (if any). If this is the case, then there only ever needs to be one "active" application profile and licence model, and these will be applied to all collections by default.

However, some customers will need to manage collections of resources which have different metadata properties, or are licensed under different sets of conditions. In this case the submission workflow will apply the application profile and the licence model appropriate to the collection the resource is in. It should be noted that:

- The application profile and licence model associated with a collection takes precedence over the application profile and licence model associated with a group (see [precedence issues](#)).
- If users are allowed to put resources into two or more collections at the same time, then the collections **must** use the same application profile and licence model (if any).

## 4.3. Managing Collections

To manage collections, click on the "Collections" link in the "Admin Tools" navigation bar.

This will open the "Collections Management" area. In this area an administrator user can change the properties of existing collections, add new collections to a system, remove collections which have no resources in them, and determine the order in which the collections will be displayed in other parts of the application.

In new installations of intraLibrary, and installations that have been migrated from previous versions of the software, there will be a single collection called "default collection". You can rename and change any other property of this collection by clicking the "Edit Properties" link for this collection.

To add a new collection, click the "Add Collection" button at the bottom of the Collections Management area. You will be taken to a form to fill in all the properties of the new collection.

For information about changing which collections a resource is in, please see the **Librarian's Guide**.

### 4.4. Managing Access

#### 4.4.1. Relationships to Groups of Users

Managing of access to collections is defined through the collections' relationships to groups of users. Every group has a relationship with every collection. For each group-collection relationship, the permission which the group members have when working with resources in a collection are defined.

To make it easier to manage the relationship between large numbers of collections and groups of users, each collection has a default set of access permissions. Groups who have not had specific access-permissions assigned to them for a particular collection get the default access permissions. Therefore the default access permissions for a collection are generally the lowest common denominator of access an administrator wants to allow.

#### 4.4.2. Access Permissions



There are five permissions associated with each group-collection relationship, and they are described below. The administrator can choose any combination of them. For example the administrator could allow members of group A to be able to find and preview resources in collection B without allowing them to export and contribute resources. The permissions are:-

- **find:** allow all users in the group to find resources in the collection using search and browse, and see the metadata for the resources they find.
- **preview:** allow all users in the group to preview resources in the collection.
- **export:** allow all users in the group to download the resource, metadata and Public URL of the resource.
- **annotate:** allow all users in the group to attach comments, star ratings and tags to resources.
- **contribute:** allow contributors in the group to add resources, edit metadata and perform other functions in accordance with their role in the group.

**Please note:** For simple resources, such as single documents or media files, the distinction between "preview" and "export" has little meaning, since it is necessary to download the file to be able to preview it. The distinction only has meaning for more complex resources, such as content packages, where it would more difficult, although not impossible, for a user to download every file connected with a resource and reassemble it.

### 4.4.3. Contributing to a Collection

For a user to contribute a resource to a collection they must:

- have a repository role of "contributor".
- be a member of a group which has contribute permission for that collection.
- have a role in the group's workflow which allows them to contribute resources.

### 4.5. Controlling Remote Access to Collections

The ability of external systems to search an intraLibrary repository is configurable for each collection within the repository. Two approaches are currently supported:

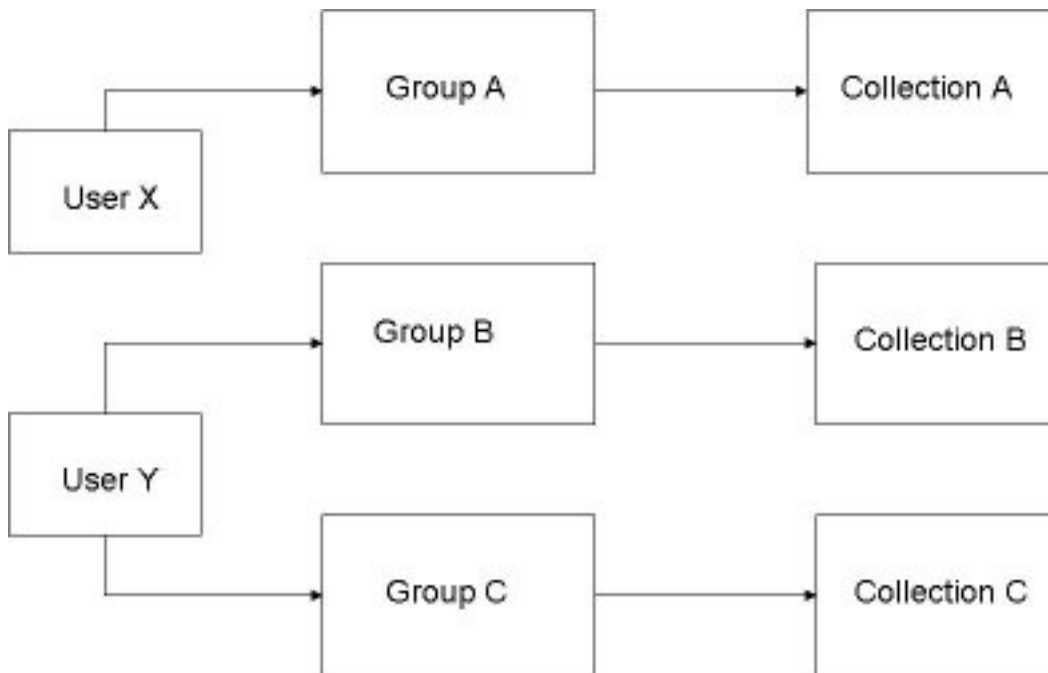
1. selectively opening/closing collections for external search
  - this is controlled by editing the properties of a collection in the "admin tools" interface
  - a collection is either open to external searching or closed to external searching (default)
2. controlling access using authentication tokens



- tokens are created and assigned to a collection in the "collections" section of the admin interface
- tokens can be supplied by external systems making search requests
- results will only be returned from collections to which the supplied token has been assigned, or to which no tokens have been assigned.

## 4.6. An Example of Using Collections

A simple example of relating groups to collections is shown in the diagram below. In this case User X is in group A and User Y is in groups B and C. Group A has full access permissions (i.e. find, preview, export, comment and contribute) to collection A but no access permissions to Collections B and C. Similarly group B has full access permissions to Collection B but no access permissions to Collections A and C.



As a practical example consider the process involved in creating a collection (Collection A) and giving full access to it for members of group A and no access to it for members of group B and group C. Assuming that groups A, B and C already exist, you would need to perform the following steps:-

- Click on the *collections* part of the admin tool bar. Now click on the *add collections* and

create a collection (calling it collection A). Unless there is a reason for not doing so give the collection access to all classification systems in the repository. Now save the collection and return to the collection management page.

- From the collection management page click on *Manage Access* for collection A.
- For group A *Use Default Permissions* should have a tick and the boxes "find", "preview", "export", "comment", and "contribute" all have a tick in them (not highlighted). This will allow a member of group A to have full access permissions to group A.
- For group B click on *Use Default Permissions* and then click the "find", "preview", "export", "comment", and "contribute" checkboxes to remove the ticks, then click "Save Permissions". This will mean that members of group B will have no access permissions for collection A.
- For group C click on *Use Default Permissions* and then click the "find", "preview", "export", "comment", and "contribute" checkboxes to remove the ticks, then click "Save Permissions". This will mean that members of group C will have no access permissions for collection A.

### 4.7. Precedence Issues

When contributing a resource to intraLibrary, there is an order of precedence which determines which application profile and license model is applied to a resource. This is only an issue in installations when more than one application profile or license model is being used.

The general order of precedence is collections, groups, default settings. This means that if features are defined at the level of collection then they will be used. If they are not defined at collection level then the ones at group level will be used and if they are not defined at group level then the default settings will be used. For example:-

- If a resource is entered into a collection which uses an application profile then this application profile will always be used.
- If a resource is entered into a collection with no application profile but the user is in a group with an associated application profile then this application profile will be used.
- If a resource is entered into a collection with no application profile and the group does not have an associated application profile then the default application profile will be used.

## 5. Metadata and Vocabularies

## 5.1. Introduction

There are several options for configuring the use and behaviour of metadata and vocabularies within intraLibrary. The most important is the "application profile", which is used to control which metadata fields are in use in a particular context, and the rules for applying these metadata fields. It is also possible to modify the metadata vocabularies available within the system. Subsets of the available metadata fields can be created to define which fields contributors and cataloguers see in a workflow.

## 5.2. Configure Application Profiles

The administrator's choice of application profile affects which metadata fields users are exposed to users, and what rules are applied to these metadata fields. Typically, an organisation or project will have well defined rules for how metadata should be used, and these are expressed using application profiles. They may well base their application profile on a published application profile, such as those given below.

Application profiles can be used to combine fields from more than one metadata model. The IEEE LOM model is included by default. If any other models have been added to the system, fields from these models will also be available for editing in the application profile editor. If you need to add additional metadata models to the system, you can do so by adding a new application profile or copying then editing an existing one in the "metadata" section of the "admin" menu. More information about defining metadata models is provided in the **Configuration Guide** for intraLibrary.

One or more application profiles can be *active*, which means they can be applied to collections or user groups. See the section on [precedence issues](#) for more information about how these two uses of application profiles are related. This enables an organisation to use different application profiles for different kinds of resources or services, whilst managing them all in a single repository system.

The application profile controls several aspects of each metadata field, including cardinality, optionality, visibility, editability and vocabulary.

IntraLibrary comes with three pre-configured application profiles:

- IntraLibrary: This is a simple application profile designed to get you started

- CanCore: The Canadian Core application profile (<http://www.cancore.ca>)
- UK LOM Core: The application profile defined by JISC CETIS and other partners (<http://www.cetis.ac.uk/profiles/uklomcore/>) as a starting point for LOM-based solutions in the UK

An administrator can also add, copy, delete and edit application profiles inline. If an application profile is to be based on one of the existing ones listed above, or another pre-defined application profile, it makes sense to use "copy" to avoid having to create your new application profile from scratch. When you click "add an application profile", the properties of fields in the new application profile are populated from the defaults that intraLibrary uses.

Intrallect can help you define an application profile that meets the specific needs of your project or service. Please contact Intrallect ([support@intrallect.com](mailto:support@intrallect.com)) for more information.

### 5.3. Edit an Application Profile

IntraLibrary includes the ability to edit an application profile inline. To do this, click the *edit* link of the relevant application profile in the listing. Editing an application profile should only be undertaken by someone who knows what they want to achieve with an application profile and who is familiar with the IEEE LOM metadata model.

One simple property of an application profile that can be changed easily is the *default optionality* of an application profile. This affects the list of metadata fields that appears by default in the metadata editor. The default is to show only the mandatory fields. If for example it was desired to show contributing users all the mandatory and recommended fields by default, the default optionality should be changed to *recommended*. It is also possible to change the names of the levels of optionality, e.g. you can set "mandatory" to appear in the Metadata Editor as "fields that must be filled in", etc.

The following properties of an individual field can be modified:

Field Size:

How many characters are allowed for an entry in the field

Cardinality:

How many instances of the field are allowed

Visibility:

Whether or not the field is displayed in the metadata editor

Editability:

Whether or not a user is able to edit the value of the field

Optionality:

Whether the field is *mandatory*, *recommended*, *optional*, *not-recommended* or *not-used*. Only the optionality levels used in an application profile appear as options to users, so if only Mandatory, Optional and Not Used are applied to fields, then contributors will only see those three options. It is also possible to change the names of the levels of optionality, e.g. you can set "mandatory" to appear in the Metadata Editor as "fields that must be filled in", etc.

Vocabulary:

Which vocabulary is used to populate a fixed vocabulary field. Select the vocabulary name from the drop-down list.

### 5.4. Manage Vocabularies

The administrator can manage the vocabularies used in the metadata by going to the "vocabularies" section of the "admin" tools. It is possible to create, edit and delete vocabularies. Vocabularies can be modified in various ways including:

- Changing the name and ID of the vocabulary
- Adding and deleting terms
- Importing terms defined in IMS VDEX format
- Changing the order the terms are displayed in
- Editing the term or "token" itself
- Changing the "source" for a term
- Creating labels for each term in multiple languages

The system will prevent you from deleting a vocabulary that is currently being applied in an application profile. It will not stop you from modifying any vocabulary, so it is important to be aware of which vocabularies are being used in the system, and how any changes will affect users.

Changes to the vocabularies affect what terms users will see in the drop-down list for each vocabulary. These drop-down lists are used in the metadata editor, and the advanced search interface. If you edit a vocabulary term (token) itself, metadata records which include the original term will not be affected. If you change the label for a vocabulary term,

this change will be visible everywhere the term is used.

### 5.5. Metadata Subsets

It is possible to control what metadata fields users in different group roles can see and edit within different processes of a workflow, e.g. you may wish those uploading resources to only see and fill in the title, description and keywords for a resource, with cataloguers filling in the rest of the metadata at the next stage. You may also wish the cataloguers to be able to see the automatically generated identifier for a resource, but not be able to edit it. An administrator can manage this by configuring metadata subsets in the "metadata" section of the "admin" tools. Metadata subsets configured there are associated with workflows by their name, so it is important that the name of the subset be exactly the same as the subset name in the associated workflow. You must configure any subsets associated with a workflow **before** you upload the workflow to the repository in the "workflows" section of the "admin" tools. However, it is possible to edit a subset once both subset and workflow are active in the repository.

## 6. Search

### 6.1. Configure Search

This interface allows you to configure the following aspects of the search features in intraLibrary.

- Simple Search: Which metadata fields are included in a *simple search*.
- Advanced Search: Which metadata fields appear in the drop-down list of available fields in the *Advanced Search* interface.
- Similarity Search: Which metadata fields appear in the drop-down list of available fields in the *Similarity Search* option next to any resource.
- Reference Search: Whether or not the contents of a field are displayed as a search link in the "view metadata" page. Typically used for "reference" fields, such as identifiers.

#### 6.1.1. Simple Search

By default intraLibrary is configured to use all the main free-text metadata fields in a simple search. Those metadata fields with fixed vocabularies are excluded from this index because

they might skew the search towards words that are used as tokens in the vocabularies.

You can use this interface to change these defaults. If you do change these settings, you must [re-index the metadata](#) before they will take effect.

### 6.1.2. Advanced Search

The default drop-down list of metadata fields in the advanced search interface includes the metadata fields that are commonly considered useful. However, you may wish your users to have access to searching other metadata fields. This interface allows you to configure which metadata fields appear in the drop-down list.

### 6.1.3. Similarity Search

All resources have a similarity search icon next to them in any search/browse result listing. The metadata fields configured appear in the list of fields as options for running a similarity search. The similarity search is run against the same field as that selected, so a title search will run an advanced search of the title field, using the terms in the original resource's title.

### 6.1.4. Reference Search

Values stored in specified metadata fields can appear as links in the view metadata page. A user can click these links to see a list of resources which have that same value in their own metadata. A good example of where this is useful is when one resource makes a reference to another resource, using the "relation" fields from the IEEE LOM metadata. The user can click on the link to the reference and see the resource the metadata refers to.

Making a metadata field available for "reference search" means a search link will be displayed for every instance of that field that appears in the view metadata page. End users can then click that search link to carry out a reference search when viewing metadata for a resource. Reference search is carried out as a "simple search" of the term in the link being clicked on, which means the search runs against all the fields specified for "simple search", not just for that specific metadata field. So you must ensure that the correct fields are configured for simple search for your users.

### 6.2. Public Saved Searches

IntraLibrary includes the ability for users to save search criteria from the *Advanced Search* page. Administrators of the system have an additional power to create *public* saved searches, for use by others users.

Public saved searches commonly define the sort of search constraint that users might want to deploy as their *default saved search*. Users will see a list of public saved searches in the relevant drop-down list in their user preferences page. They can select a default saved search to constrain the results of all of their search and browse operations.

Contributor users can use public saved searches or their own personal saved searches to search the contents of their [work area](#).

NB: Public saved searches are not currently exposed to other users in the advanced search interface itself.

## 7. Agreements

The organisation running intraLibrary may wish to use the usage agreement feature of intraLibrary to record users' acceptance of a general agreement to use the system. This agreement may encompass the legal agreements that are defined for the community licences described in the next section or it may simply be a code of conduct. The text of a usage agreement can be entered into the system by the administrator. On first or every login, the user can be presented with this agreement to accept or reject. Details of the acceptance of an agreement are recorded in the database, and users can see what agreements they have accepted at any time in their profile.

## 8. Licences

### 8.1. Introduction

A common barrier to the reuse of resources is the issue of ownership and copyright. Conflicting copyright agreements, or uncertainty over ownership and usage permissions, is



a constant thorn in the side of those who want to share resources within or between organisations.

Intrallect has included a community licensing solution in intraLibrary to help overcome some of these barriers. This solution enables resources to be deposited in intraLibrary under one or more community licences. These licences define the conditions under which the resource is made available to other users. One example of such a set of licences is [Creative Commons](#).

It is also common to want to have some kind of usage agreement for a repository. Some users will be depositing resources, others will be exporting and reusing resources. Organisations will often want these users to acknowledge that certain conditions apply to their use of the repository. IntraLibrary administrators can define usage agreements for the various roles in the system, which users must accept before they get access to the relevant functions.

### 8.2. Licensing Model

An organisation can define its own licensing model to be used in their intraLibrary repository. The licensing model consists of a set of one or more community licences. Each licence can have a different combination of the various parameters available in the Intrallect Licensing Information Model (ILIM), which is based on the Open Digital Rights Language (ODRL). These licences define the conditions under which the resource is made available to other users.

The licences can include permission, requirement and constraint parameters such as:

- (Can) modify
- (Must) attribute
- Non-commercial use (only)

There is also a transfer permission, which specifies how strictly the above conditions are applied if the resource is passed on to other users. This allows licence conditions of the *sharealike* type, as used by Creative Commons, to be expressed.

### 8.3. Depositing Resources

A *licence editor* action can be included in any process of a submission workflow for an intraLibrary repository. This gives the user an opportunity to select the licence under which they make their resource available. The completion criteria of the workflow process may define a subset of licences to which the resource must conform.

The licence editor enables the user to make a choice between the available licence templates. Each licence may be presented as a row of icons or phrases with an associated radio button. The user clicks the radio button next to their chosen licence. They are then shown other fields that may need to be populated for that instance of the licence, such as the identity of the copyright holder(s). When the user has saved all the necessary information, the licence action has been completed.

### 8.4. Accessing Resources

When the community licensing system is activated, the licence properties of resources are visible to users when searching and browsing. When a user sees a list of resources, the type of licence will be indicated to them using a set of symbols or phrases, or alternatively, a single Terms and Conditions of Use link may be shown directing them to an external web page. The symbols or phrases represent a particular permission or constraints on the user of the resource, which may differ between the available licences. Each symbol or phrase can be a link to a description of the relevant constraint, or the relevant part of the full text of a licence agreement.

It is possible to associate different licensing systems with different collections in intraLibrary.

### 8.5. Configuring the System

For an organisation to apply their own set of licences in intraLibrary, they will need to do several things.

1. Discuss possible licensing models with their user community and legal advisors
2. Select or define one or more licences by:
  - mapping these licences against the parameters of the Intrallect Licence Information Model (ILIM) which is based on the Open Digital Rights Language (ODRL)
  - choosing or commissioning formal licence agreements

3. Select or define some icons which represent the distinctive conditions of the chosen licences
4. Arrange for their licence agreements to be made publicly available, e.g. through a persistent URL
5. Work with Intrallect to implement these definitions in intraLibrary (See the **Configuration Guide** for more details.)
6. Once everything is configured within the system, they must switch on the "enable rights mode" option in the "system" menu to make licensing options appear to users.

### 8.6. What Community Licensing Doesn't Do

The community licensing solution in intraLibrary does not:

- Enable users to constrain searches for resources by licence criteria
- Enable administrators to control access to resources, such as
  - Restrictions on user access by licence type
  - Restrictions on domain/institution access by licence type
- Record the user's acceptance of the licence for each resource
- Include any kind of technical enforcement of licence conditions.

### 8.7. Technical Solution

The licence information needs to be stored in the repository, and potentially be portable between repositories and other systems. IntraLibrary uses the following approaches to storage and exchange of licence info:

- Store licence information for each resource according to a licence information model, ILIM, based on ODRL
- Allow licence information to be harvested by other systems, following the approach of [Project RoMEO](#)
- Export and import licence information in IMS Content Packages following the model established by the [COLIS project](#)
- Put a link to online copies of the legal agreements in the relevant fields of other metadata formats, such as:

### 8.8. References on Licences

- Creative Commons: <http://creativecommons.org/>
- Project RoMEO: <http://www.lboro.ac.uk/departments/ls/disresearch/romeo/>

- Open Digital Rights Language (ODRL): <http://odrl.net/>
- COLIS Project: <http://www.colis.mq.edu.au/>

## 9. System

The *system* section has five tabbed sub-sections:

- User Options.
- Metadata Options.
- RSS Options.
- System Options.
- Tech Summary.

The options within these are detailed below.

### 9.1. General Configuration

There are a number of configuration settings which can be controlled from the tabbed sub-sections within the System tab.

#### 9.1.1. User Options

The following settings can be configured for users:

- The ability to delete and move resources which they have contributed to the library. Set to "no" by default; in this case only users in the librarian role may move or delete resources.
- The ability to request a new password to be sent to them. A "forgotten password?" link will appear in the login screen if this is enabled. Set to "no" by default.
- Whether or not users can choose not to use a metadata template. Setting this to "yes" means that a user could choose to circumvent your default metadata template; if this template is generating important fields such as identifiers, resources uploaded by such a user would have no identifiers.
- Whether or not librarian and/or administrator users receive email alerts when any resource receives a comment or tag from a user.

#### 9.1.2. Metadata Options

The following metadata settings can be configured:

- Whether or not the contributing user's default metadata template should be applied to a content package on import. If this is set to "no" the package will not have any local macros applied, such as those generating unique local identifiers.
- Whether or not users' comments are exported with metadata. This applies only to comments, which are part of standard LOM metadata. Star ratings and tags are local extensions of the LOM annotation field and so would not be exported in any case.
- The metadata template (if any) to apply to the results of an external SRU/SRW search request (see the Integration Guide for more information).
- The metadata template (if any) to apply to the results of an external OAI-PMH harvest request (see the Integration Guide for more information).

### 9.1.3. RSS Options

The following RSS newsfeed settings can be configured:

- The location of a stylesheet to be used for RSS newsfeeds.
- The frequency with which the RSS cache is updated.

### 9.1.4. System Options

The following general system settings can be configured:

- The default language for the system.
- Which licence pattern (metadata template for ODRL rights metadata), from any of the system's licensing models, will be applied by default to resources on upload. See [Licences](#) section.
- The number of records that should be returned for a single OAI list request. The default is 1000.
- The maximum basket size below which a user can export their entire My Basket contents. The default is 100MB.
- Whether or not metadata at any level of a content package is searched. The default is "No", in which case only metadata in the package manifest is searched.
- The URL for your local intraLibrary installation (used for generating Public URLs and so on, when a user hasn't logged in, e.g. when SRU/SRW searching is used).

## 10. Support Login

### 10.1. When You Need Support


If you have a problem or bug in your installation of intraLibrary, it may occasionally be necessary for a member of Intrallect's support team to get access to your repository. You can allow this by switching on "support access".

### 10.2. What Support Access Is

Enabling support access will create a temporary user with admin rights and send the login details to the Intrallect support team. This will grant the Intrallect support team temporary access to the application in order to diagnose the problem. The support user will persist until the support access is disabled.

In order for support access to work, the application must have enough capacity in its software licence to allow the addition of a new (temporary) user with contributor, librarian and administrator permissions.

### 10.3. Activating Support Access

In normal operation, the support section of "Admin" will include the message "Support access is currently disabled", and display an icon  to "Enable External Support Access". If you have been asked by a member of the Intrallect support team to activate support access in your installation of intraLibrary, click the icon to "Enable External Support Access".

If you do not have enough capacity on your software licence, the page will display the message "it was not possible to enable support access as there was not enough capacity on your licence." If this happens, you need to take away contributor privileges from one of your existing users before you can enable support access.

### 10.4. De-Activating Support Access

When support access is active, the support section of the "Admin Tools" will display information about the status and identity of the support user, and the icon to "Disable External Support Access".

If you click the icon to "Disable External Support Access", then the support login will be de-activated. Re-activating support access again will create a completely new login for the Intrallect support team.

## 11. Mail Configuration

### 11.1. Email Basics

When the intraLibrary system sends out emails, it uses the Simple Mail Transfer Protocol (SMTP). To do this, it must be able to connect to an SMTP server. An SMTP server accepts emails from sources it recognises, and relays the emails towards their destination via a chain of similar SMTP servers.

The SMTP server you use may be on your local network, or available on the internet. A non-local SMTP server will normally require a username and password before relaying emails.

### 11.2. Configuring the SMTP Settings

The following settings are required for a non-authenticated SMTP server:

- Server Description: Anything you want to use to identify the SMTP server
- Server Name: The full domain (or IP) address of the SMTP server
- Server Port: The IP port used to connect to the SMTP server (usually 25)

The following additional settings are required for an authenticated SMTP server:

- Username: The username for authentication by the SMTP server (if any)
- Password: The password for authentication by the SMTP server (if any)

The values you enter in these fields may well be the same as the configuration of your outgoing mail server in your normal mail client, e.g. Mozilla Thunderbird or Microsoft Outlook.

### 11.3. Testing Mail Settings

It can sometimes be difficult to be sure if you've got the SMTP settings right. The system

can send out a test email to help you check everything is working.

To check the email connection is working:

1. enter your email address
2. click "Send Test Email"
3. check your email - you should receive an email with the return address that has been assigned to your intraLibrary installation, with a subject heading of "Testing IntraLibrary SMTP Server"

If you don't receive an email, first check your spam or junk folders to see if it ended up there. If not, then you need to go back and review your SMTP settings.

### 11.4. Publication Notification

Email addresses to which notifications are sent whenever a resource is published.

## 12. Metadata Reports

In the "reports" section of the "admin" tab, you can generate metadata reports in Excel or CSV format. These metadata reports list the values contained in selected metadata fields. Report configurations can be stored and reused as templates.

### 12.1. Configuring Metadata Reports

The following variables may be configured for reports by filling in the "Generate Metadata Reports" form:

- **Report name:** A title for the report, which will be used in the Report Templates list in the left-hand pane to identify the report.
- **Report filename:** The filename for the exported Excel or CSV file.
- **Metadata fields:** Which metadata fields to include in the report.
- **Multiplicity of metadata fields:** Maximum number of entries per metadata field. For instance, for the metadata field "1.6 Keyword", where a multiplicity of 5 is selected, the report will include the first five keywords for each resource reported on, with superfluous cells left blank where there are less than five keywords present.



- **Language:** Which language you want the report to include. For instance, if you have metadata records in both English and Spanish, you can select whether the report will include English or Spanish values for the selected fields.
- **Collections:** The report can include metadata from resources held in any or all collections in the library.
- **Published or unpublished?:** The report can include metadata from unpublished resources only, published resources only, or both.
- **Report dates:** Metadata will be included in the report from resources uploaded, last modified, or both, between the dates selected. Leaving the date fields blank means that resources uploaded or last modified at any time will be included.
- **Report format:** Whether the report is to be exported in Excel or CSV format.

To create a report:

- Fill in the "Generate Metadata Reports" form.
- Click "Submit". Your browser will give you the option to open or save the report.
- You can clear the form at any time by clicking "clear report form" in the left-hand pane.

### 12.2. Saving Metadata Report Templates

To save a report configuration as a template:

- Fill in the "Generate Metadata Reports" form.
- Click "Save". The report template will then appear listed in the left-hand pane.
- To reuse a report template, click the "reuse/edit template" icon. The report form will be immediately filled in with your pre-selected configuration. You can use it exactly as is or change any part of it before clicking "Submit". To save any changes to the template click "Save" again.

### 12.3. Repository Statistics Table

Displays a range of basic statistics about your local intraLibrary installation, including the total number of resources; the number of published and unpublished resources; the total number of users; the number of users in each role (i.e. guest, contributor, librarian and administrator); the number of groups and collections; and the number of users currently logged in ("Current user sessions").

This table also gives you the version and build number of intraLibrary currently being used for your installation.

## 13. Maintenance

### 13.1. Resource Cache

IntraLibrary uses a memory cache of resources to improve performance and reduce the load on the underlying database. The maximum number of resources stored in the cache is configurable. A larger number will generally improve performance if sufficient memory is available on the server. See the installation notes for more details.

Occasionally a resource may get corrupted in the cache. For example, all its metadata might appear to have been lost. If this happens, the cache can be rebuilt from the resource data stored in the database by clicking the *rebuild resource cache* button.

### 13.2. Search Index (Re-index Metadata)

If a search in intraLibrary seems to be producing strange results, or is missing resources you think it should be finding, the first course of action should be to re-index the metadata. It's also important to do this if the search configuration is modified.

Re-indexing rebuilds the index the search engine uses to find resources that are relevant to a given query. Depending on the number of resources in the system, re-indexing could take a few seconds, a few minutes, or much longer. Re-indexing updates one metadata record at a time, so there is no major problem in re-indexing whilst users might be searching for resources.

### 13.3. Image Thumbnails

If your installation is configured to support thumbnailing of image resources, you will see a button to "rebuild image thumbnails". Thumbnails will be created for all single (JPEG) images when they are uploaded into the repository. See the installation guide for more information and configuring the repository to support thumbnailing.

Occasionally it may be necessary to rebuild thumbnails for all the image resources. You might need to do this for the following reasons:

- You have just activated thumbnailing in a repository that already contains some image resources.
- You have changed one or more properties of the thumbnails, such as the thumbnail dimensions (see the intraLibrary **Installation Guide** for more information on how to do this).
- You can see some corrupt thumbnails appearing in the search results. In normal circumstances, this should not happen.

If you need to rebuild thumbnails for any reason, click the "rebuild image thumbnails" button. Rebuilding the thumbnails for a large number of image resources can take a while. Whilst thumbnails are being rebuilt, the "rebuild image thumbnails" button will be greyed out every time you visit the "system" page.

A message below the "rebuild image thumbnails" link tells you what the current thumbnails settings are. For example it might say:

- *Current thumbnail dimensions are 75px square for small thumbnails, 200px wide for large thumbnails and are of quality 7/10.*

### 13.4. Metadata Templates

If you have made changes or additions to the [metadata templates](#), you will need to use the *Reload templates* link to reload the templates before your changes take effect.

## 14. Metadata Templates

### 14.1. Introducing Metadata Templates

To save the user having to repetitively enter easily definable metadata values, intraLibrary can optionally populate metadata records with pre-defined values. These pre-defined, or default, values are defined in stored *templates* which must themselves be valid IMS Metadata records in XML format. In these *metadata templates* any metadata field can be assigned a default value, or multiple default values where appropriate.

### 14.2. Creating Metadata Templates

Metadata templates are defined using XML files which follow the IMS Learning Resource Metadata XML Schema. These files must be placed in the *config* directory of your intraLibrary installation, inside a folder called *metadataTemplates*. An easy way to create a metadata template is to create a dummy resource in intraLibrary with the metadata properties you want. You can then export an IMS Learning Resource Metadata record from this resource, and use the resulting XML file as the basis for your new metadata template. The filename of the metadata template must be of the form *templatename.xml*.

When you have added or updated templates in the *metadataTemplates* folder, you need to click *Reload Templates* in the *maintenance* section of the intraLibrary admin tools to update the templates used inside intraLibrary. Templates will appear in the relevant drop-down list in the user preferences page. The name of each template will be the name of the file with the *.xml* suffix removed. There is a template called *default* that is supplied with the system. You can overwrite this by creating your own *default.xml* template in the *metadataTemplates* folder. Templates are also made available to users in the Metadata Editor in their work area, so that they can be applied while editing metadata.

### 14.3. Service Result Templates

The system can also be configured to apply metadata templates to records exposed to external systems via service interfaces. You may wish, for example, to ensure that a particular copyright statement appears in metadata records harvested via an OAI-PMH service, or that a particular field is populated with a Public URL when using SRU to query your repository from your website. More information about these interfaces can be found in the intraLibrary **Integration Guide**.

The following options are available in the [Metadata Options](#) tab of the [system](#) page:

- Metadata template to apply to search results
  - values and macros in the selected template will overwrite the values stored in the relevant fields of each record returned in a search request.
- Metadata template to apply to harvested results
  - values and macros in the selected template will overwrite the values stored in the relevant fields of each record returned in a harvest request.

### 14.4. Template Format

Metadata templates are defined in XML files which follow the IMS Learning Resource Metadata XML Schema. Any property of the IMS/IEEE LOM metadata model can be defined in a metadata template. The metadata fields in a template can be populated either by static strings, or dynamic macros generated using information available to intraLibrary.

Metadata template macros can be called up using the macro name in curly brackets, such as {UserFullName}. A complete list of metadata template macros is given in [Appendix 1: Metadata Template Macros](#). Metadata template macros cannot currently be called up as part of another string, so including *The user who wrote this is {UserFullName}* in the *copyright.description* field won't work.

The following example shows how you can mix static strings and metadata template macros in the lifecycle section of a template defined IMS Learning Resource Metadata (XML) format:

```

<lifecycle>
  <contribute>
    <role>
      <source>
<langstring xml:lang="x-none">LOMv1.0</langstring>
      </source>
      <value>
<langstring xml:lang="x-none">Author</langstring>
      </value>
    </role>
    <centity>
<vcard>{UserVCard}</vcard>
    </centity>
    <date>
<datetime>{NowCurrentDate}</datetime>
    </date>
  </contribute>
</lifecycle>

```

You can use an XML Editor to compose and validate your metadata template. Using some

of the metadata template macros in particular fields may cause a validation error, but this does not necessarily mean your template will not work in intraLibrary. For example, using the {ObjectFileSize} macro in the technical.size field of the metadata will cause a validation error, because the XML Schema requires this element to contain a number. Therefore, if you are using a validating editor such as XML Spy, it is probably best to insert the metadata template macro after you have finalised and validated the overall structure of your template.

## 15. Email Templates

There are six different types of email alerts which can be sent by intraLibrary. These alerts are used to notify users about changes to resources such as when a resource has passed from one stage to another, when a resource has been published and when a resource has been annotated.

### 15.1. Email Alerts for Stage Change

There are three types of alerts that can be used to notify users about resources moving from one stage to another. They are:

- Letting individual users know about state changes to a resource:  
An individual user can choose to receive an email when a resource enters a stage they have access to. This option is chosen by the individual user from the groups section of their profile page. The email template for this alert is StateChangeEmail.txt which is stored in the email templates directory in the configuration directory on the intraLibrary installation server.
- Letting external people know about state changes to a resource:  
When writing the xml for a workflow, it is possible to include an action "Email addresses to alert" followed by email addresses and alerts will be sent when the stage is completed (see section on workflows for more details). The template for this alert is StageChangeExternalEmail.txt which is stored in the email templates directory in the configuration directory on the intraLibrary installation server.
- Letting the resource owner know about state changes to a resource:  
The workflow template also has an action to "alert the resource owner" and this will alert the resource owner when the resource changes state (see sections on workflows for more details). The template for this alert is "StateChangeObjectOwnerEmail.txt" which is stored in the email templates directory in the configuration directory on the intraLibrary installation server.

### 15.2. Email Alerts for Publication of Resource

There are two types of alerts which can be used to notify users when a resource has been

published. They are:

- **Publication notification for user:**  
An individual user can select the "send notification email" box from "preferences" on their "profile" screen. The email template for this is "UserPublishNotificationEmail.txt" which is stored in the email templates directory on the configuration directory on the server.
- **Publish notification for external user:**  
The administrator can decide who receives alerts when a resource has been published in the repository. This is entered at the foot of the "mail" section which is part of the "admin page". Any number of email addresses can be chosen to receive a message when a resource is published in a repository. The text for the message is "adminPublishNotificationEmail.txt" which is stored in the email templates directory in the configuration directory on the intraLibrary installation server.

### 15.3. Email Alerts for Annotation of a Resource

The administrator can choose to send an alert to all administrators and/or all librarians when a resource has been commented on or tagged by a user. This is done through the "system" section "user options" tab of the "admin page". The text for this alert can be altered by changing the "annotationNotificationEmail.txt" which is stored in the email templates directory on the configuration directory.

### 15.4. Configuring the Email Templates

In the above section six different types of email notifications were considered. The text for these email notifications can be modified by the administrator by altering the templates which are contained in the email templates directory on the configuration directory on the server in which intraLibrary is installed. The example below shows a sample template for an email about a resource changing state (stageChangeEmail.txt). There is a list of the macros which can be used for each template in the section on Email Template Macros.

**A new learning object is now available.**

**Click on the link below to preview the resource:**

\_\_url\_\_

**title:**

\_\_title\_\_

**description:**  
 \_\_description\_\_  
**comment:**  
 \_\_comment\_\_  
**classifications:**  
 \_\_classifications\_\_

**This message was sent to you by the administrator of intraLibrary.**

## 15.5. Email Template Macros

The following macros can be used by all the email templates (StageChangeEmail.txt, StageChangeExternalEmail.txt, StageChangeObjectOwnerEmail.txt, UserPublishNotificationEmail.txt, adminPublishNotification.txt, annotationNotificationEmail.txt)

| Variable            | Description   |
|---------------------|---|
| __repositoryName__  | Name of the repository which the resource is stored in              |
| __url__             | The public URL of the resource                                      |
| __title__           | Name of the repository which the resource is stored in              |
| __description__     | The description of the resource (from metadata field "description") |
| __currentStage__    | The stage which the resource is currently in                        |
| __currentWorkflow__ | The workflow which the resource is currently in                     |
| __classifications__ | List of all taxonpaths which the resource is classified under       |

The following variables are available for use in each email template except the annotation notification email (i.e StageChangeEmail.txt, StageChangeExternalEmail.txt, StageChangeObjectOwnerEmail.txt, UserPublishNotificationEmail.txt, adminPublishNotification.txt, )

| Variable      | Description  |
|---------------|--|
| __publisher__ | Name of the person publishing or moving the resource |
| __comment__   | Comment made when moving the resource                |



The following variables are available only on the annotation notification email (annotationNotificationEmail.txt)

| Variable            | Description  |
|---------------------|--|
| __annotation_text__ | The text body of the annotation (comment) which has been written by the user |
| __star_rating__     | Star rating  |
| __tagstrings__      | Tags   |
| __annotation_date__ | Date of the annotation   |
| __Username__        | The name of the user who created the annotation                              |

## 16. Appendix 1: Metadata Template Macros

The macros available for populating metadata templates are:

| Property                      | Description   | Possible Uses (target metadata fields)                            |
|-------------------------------|---|---|
| <b>Properties of the user</b> |   |   |
| {UserFullName}                | User Account  | The full name of the user, as stored in the user profile          |
| {UserAffiliation}             | The affiliation (employer, institution or project) of the user, as stored in the user profile |   |
| {UserEmail}                   | The email address of the user, as stored in the user profile                                  |   |
| {UserPreferredLanguage}       | The language the user has chosen as their preferred language in their user profile            | general.language<br>metametadata.language<br>educational.language |
| {UserUsername}                | The intraLibrary username of the user   |   |
| {UserVCard}                   | Returns the user's stored details in vcard  | lifecycle.contribute.centity                                      |

|  |  |  |
|--|--|--|
|  | format   | metametadata.contribute.centity<br>annotation.centity                                    |
| <b>Properties related to the current date / time</b> |  |  |
| {NowCurrentDate}                                     | Current date in UTC<br>format  | lifecycle.contribute.datetime<br>metametadata.contribute.datetime<br>annotation.datetime |
| {NowGuid}  | A globally unique<br>identifier (GUID)<br>generated using the<br>current time  |  |
| <b>Properties of the resource</b>                    |  |  |
| {ObjectMimeType}                                     | Mime type of resource<br>matched by file<br>extension  | technical.format   |
| {ObjectPublicURL}                                    | Full Public URL for this<br>resource for this user   | technical.location   |
| {ObjectFileSize}                                     | Size in bytes of<br>uploaded file or<br>package .zip file  | technical.size   |
| {ObjectAggregationLevel}                             | Not currently supported<br>Placeholder for an<br>automatically generated<br>value of 1-4 depending<br>on predefined rules for<br>assessing the<br>complexity or<br>granularity of a resource |  |
| {ObjectCreationDate}                                 | Date of creation,<br>extracted from<br>metadata in digital file.<br>Currently supported for<br>JPEG images<br>(exif:DateTimeOriginal)<br>only  | lifecycle.contribute.date  |
| {ObjectCreator}                                      | Author/creator of digital  | lifecycle.contribute.centity   |

|   |   |                             |
|---|---|-----------------------------|
|   | resource, extracted from metadata in digital file. Currently supported for JPEG images (tiff:Artist) only |                             |
| {ObjectFilename}                          | Returns filename of uploaded single file, or filename of package .zip file                                | general.title               |
| {ObjectId}                                | The OAI identifier of the resource  | general.catalogentry.entry  |
| {ObjectLocalID}                           | The number of the resource in the repository  | general.catalogentry.entry  |
| {ObjectPublicManifestURL}                 | A Public URL allowing download of the resource's content package manifest, for this user                  |                             |
| {ObjectPublicURLAsZIP}                    | A Public URL allowing download of the resource as a .zip file IMS Content Package, for this user          |                             |
| <b>Properties of the repository</b>       |   |                             |
| {RepositoryName}                          | The name given to the intraLibrary repository in the properties file                                      |                             |
| {RepositoryId}                            | OAI identifier of the intraLibrary repository assigned in the properties file                             |                             |
| {RepositoryAdminEmail}                    | Email address of the repository administrator   |                             |
| <b>Properties of application profiles</b> |   |                             |
| {ApplicationProfileName}                  | The recognised name of the application profile,   | metametadata.metadatascheme |

|   |   |                                       |
|---|---|---------------------------------------|
|   | e.g. "CanCore"  |                                       |
| {ApplicationProfileSource}                                      | The authoritative source for the application profile  |                                       |
| {ApplicationProfileVersion}                                     | Version number of the application profile   |                                       |
| <b>Properties of JPEG images (extracted from EXIF metadata)</b> |   |                                       |
| {ImageCompressionScheme}  | tiff:Compression  | Z39.87: Compression.CompressionScheme |
| {ImageCompressionRatio}   | exif:CompressedBitsPerPixel   | Z39.87: Compression.CompressionLevel  |
| {ImagePixelWidth}   | exif:PixelXDimension  | Z39.87: ImageWidth                    |
| {ImagePixelHeight}  | exif:PixelYDimension  | Z39.87: ImageLength                   |
| {ImageCameraMake}   | tiff:Make   | Z39.87: DigitalCameraManufacturer     |
| {ImageCameraModel}  | tiff:Model  | Z39.87: DigitalCameraModel            |
| {ImageOrientation}  | tiff:Orientation  | Z39.87: Orientation                   |
| {ObjectCreationDate}  | Date of creation, extracted from metadata in digital file. Currently supported for JPEG images (exif:DateTimeOriginal) only         | lifecycle.contribute.date             |
| {ObjectCreator}   | Author/creator of digital resource, extracted from metadata in digital file. Currently supported for JPEG images (tiff:Artist) only | lifecycle.contribute.centity          |