OJS/OCS Quick Reference

Version 1.0

Last updated: June 23 2008

STANFORD UNIVERSITY

SFU SIMON FRASER UNIVERSITY LIBRARY

SFU CANADIAN CENTRE FOR STUDIES IN PUBLISHING

ASU ARIZONA STATE UNIVERSITY

THE UNIVERSITY OF BRITISH COLUMBIA Faculty of Education

PKP PUBLIC KNOWLEDGE PROJECT

OJS/OCS Quick Reference

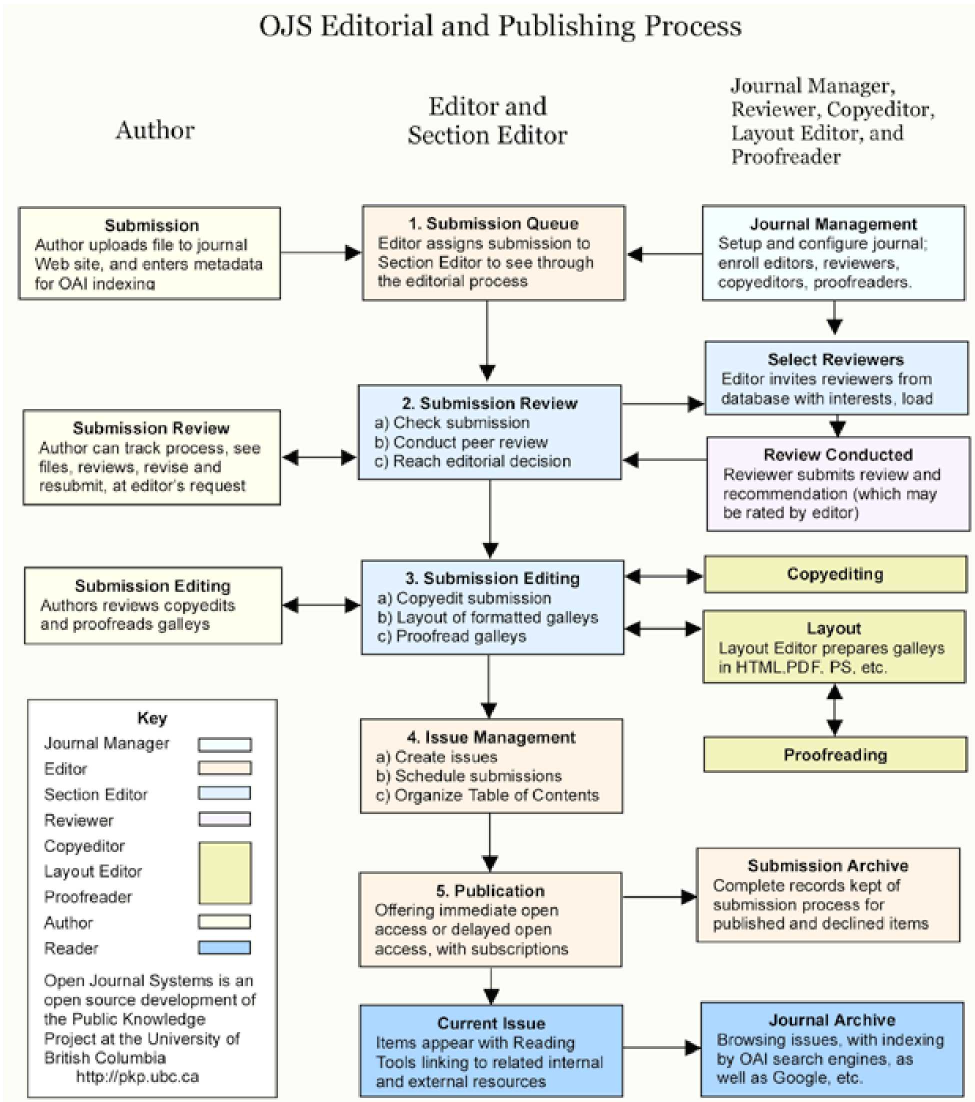
Table of Contents

Getting Help

Website: http://pkp.sfu.ca/ojs http://pkp.sfu.ca/ocs
Support Forum: http://pkp.sfu.ca/support/forum
Email: pkp-support@sfu.ca

Journal Editor / Conference Director

Open Journal Systems



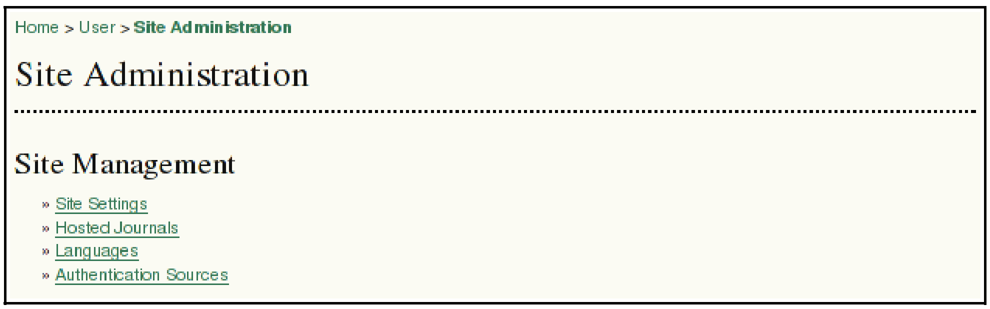
Creating Journals (Open Journal Systems)

A single installation of OJS is capable of running either a single journal or several journals. Nearly all settings and options can be configured for each journal separately, so it is possible to host distinct journals within the same installation without causing interference.

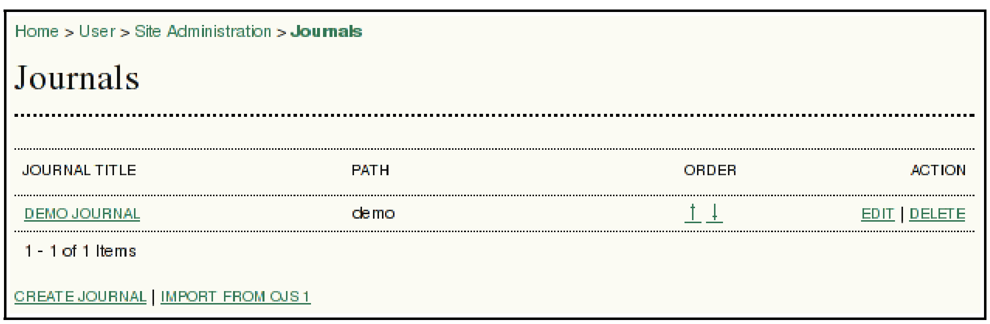
The user database is shared between journals, however, so while it is possible for users to have different roles in different journals, e.g. Journal Manager in one and Reader in another, it is not possible for users to have completely distinct profiles, e.g. names, biography statements, etc.)

The Site Administrator is responsible for creating journals within OJS. Once logged in as Site Administrator, follow these steps to create a journal:

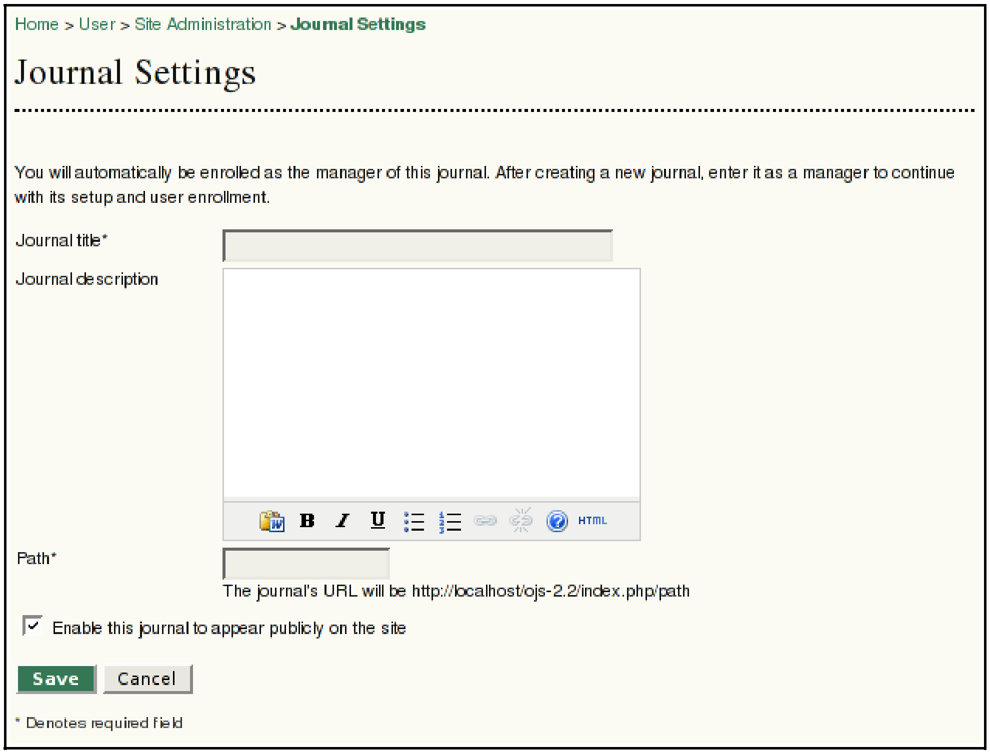
1. Find the “Hosted Journals” link in Site Administration:



2. Click on it to find a list of currently hosted journals:



3. Follow the “Create Journal” link:

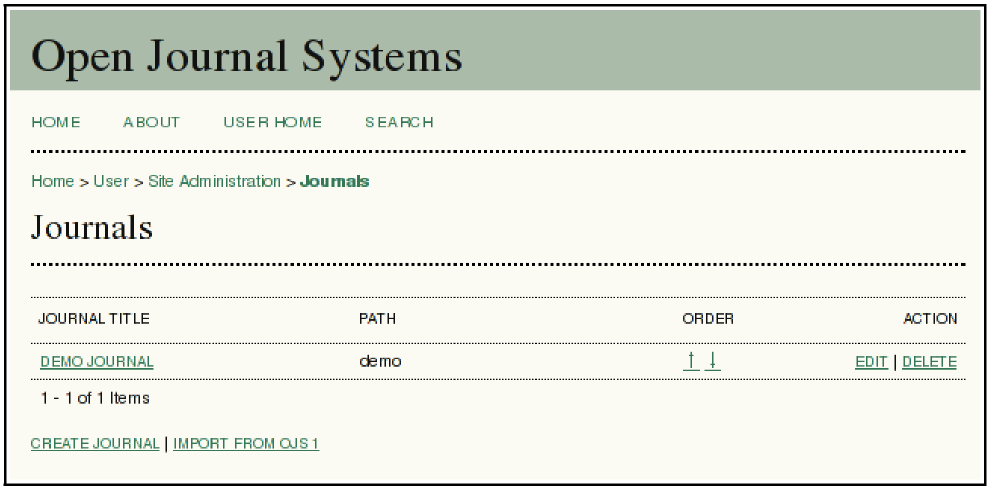


4. After the journal has been created, it is strongly suggested that the Journal Manager review all Setup pages for the new journal.

Steps to Publication (Open Journal Systems)

Requirements for Submissions

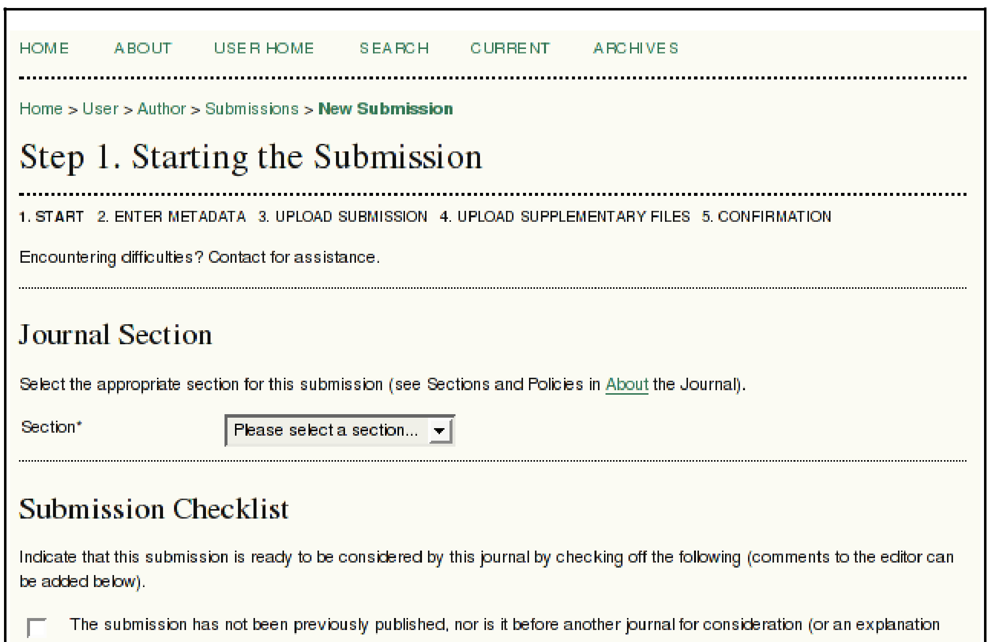
In order for submissions to be accepted, you will need to have created a Journal:



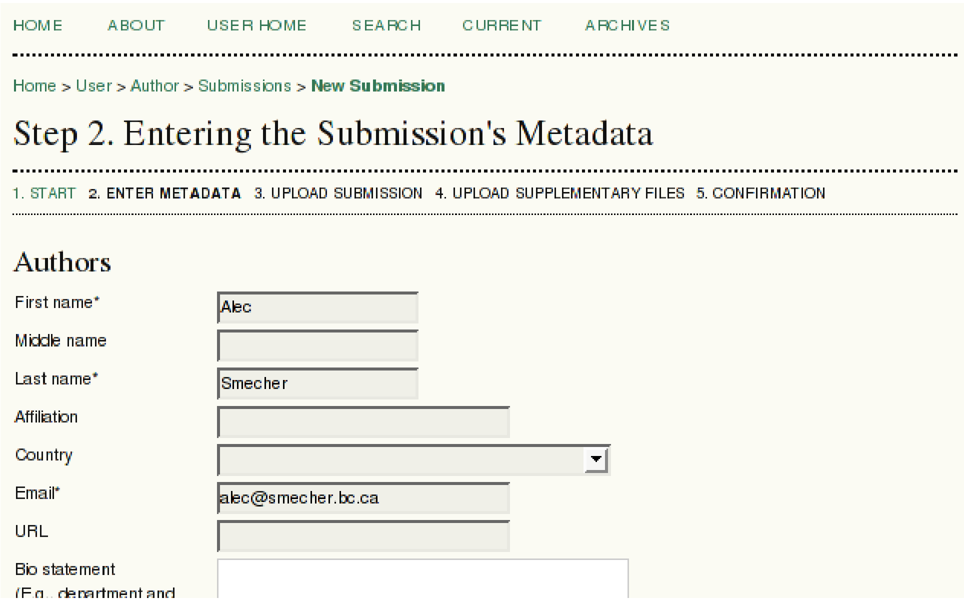
Submission to Publication

The following list describes the steps that a submission must undergo to reach publication.

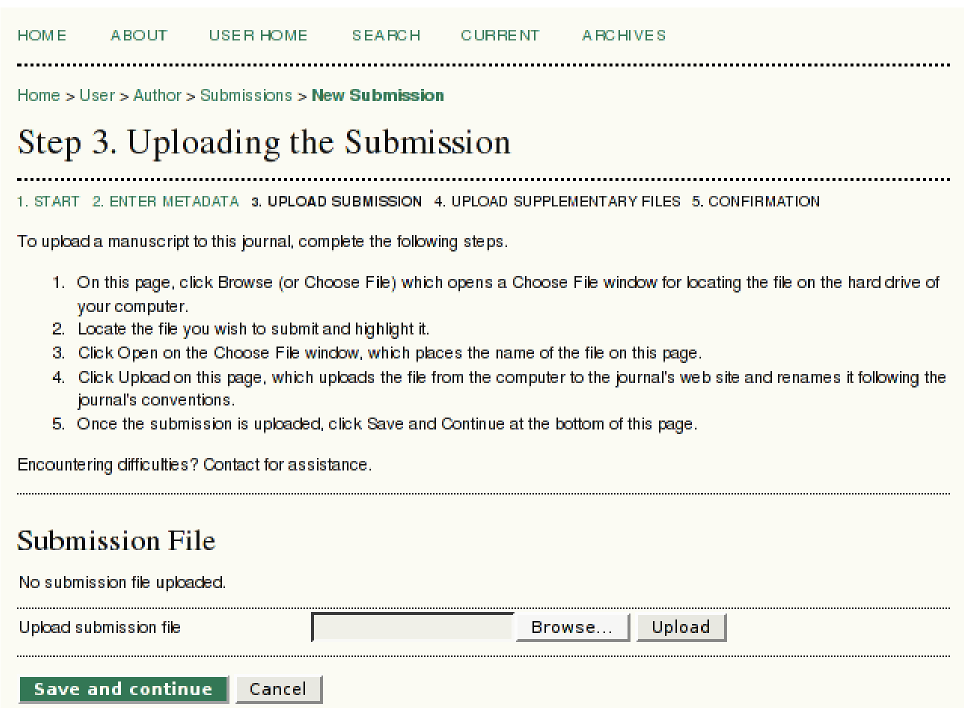
1. **Author Submits Article.** The article submission process typically consists of 5 steps:



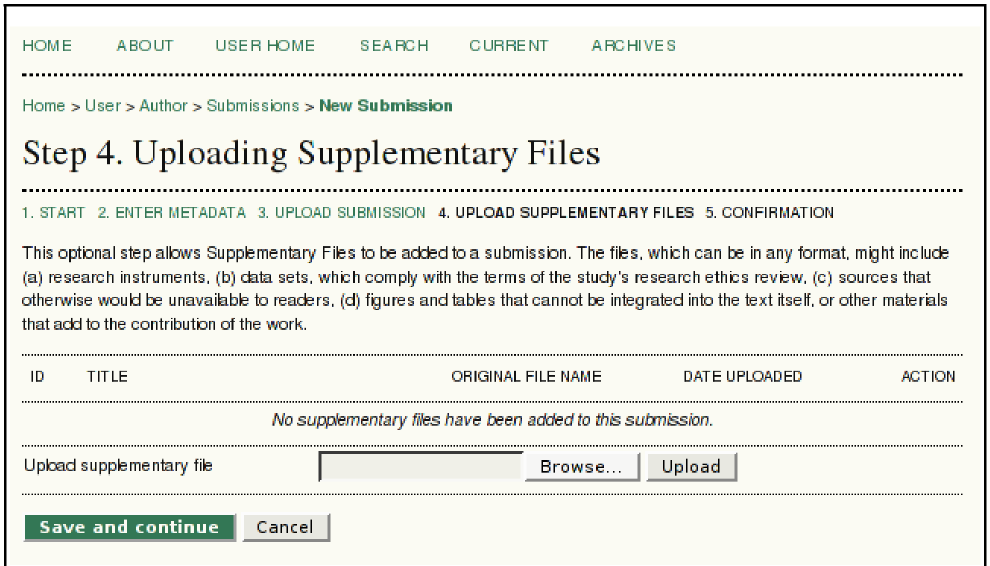
Submission process, step 1



Submission process, step 2



Submission process, step 3



Submission process, step 4

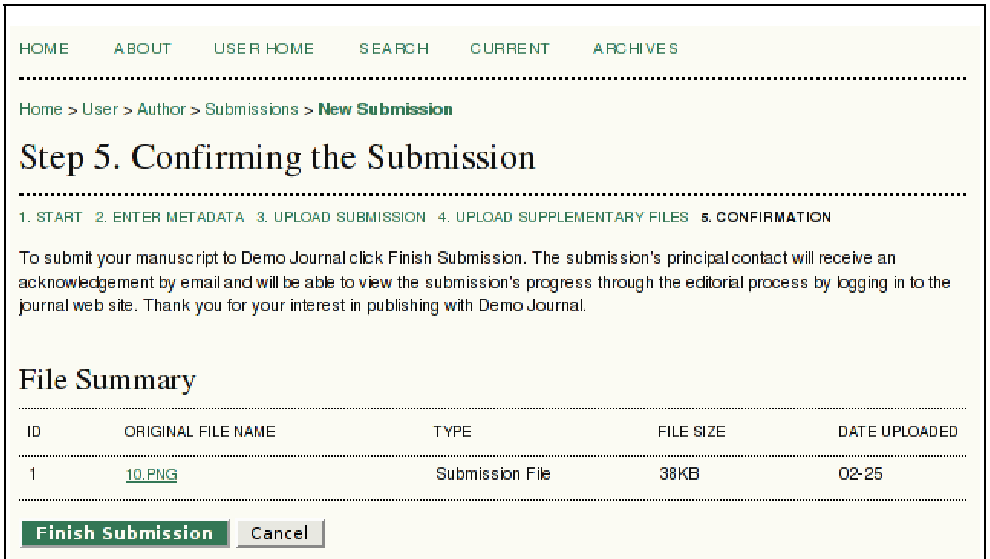
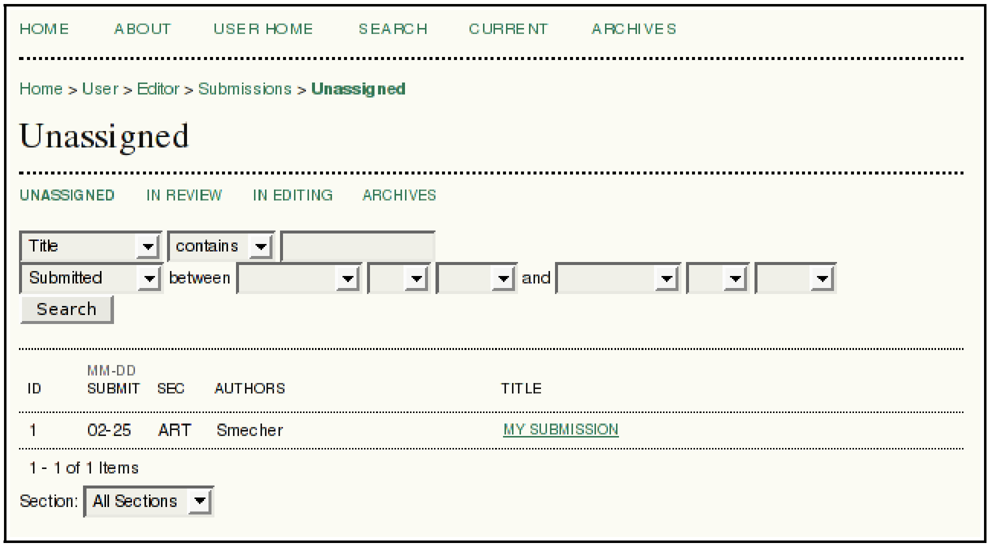


Illustration 1: Submission process, step 5

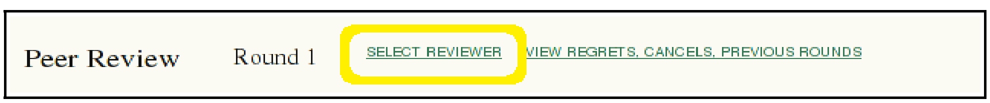
2. **Editor Receives Submission.** Depending on the journal's configuration, once the submission has been completed by the Author, it will appear in the Director's “Unassigned” or “In Review” list.



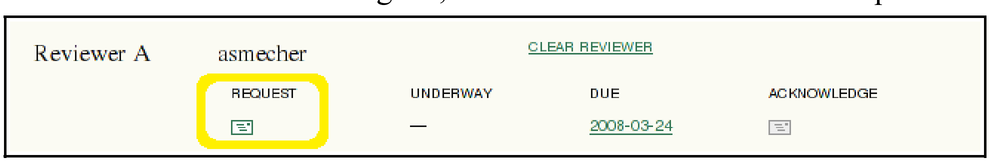
Click the title to and find the submission's Summary page. If no Editors or Section Editors have been assigned yet, assign one under the “Editors” heading:

Assigning an Editor or Section Editor will move the submission from the “Unassigned” list into the “In Review” list.

3. **Editor Initiates Peer Review.** Find the submission's “Review” page and use the “Select Reviewer” link to choose the reviewers you wish to assign:



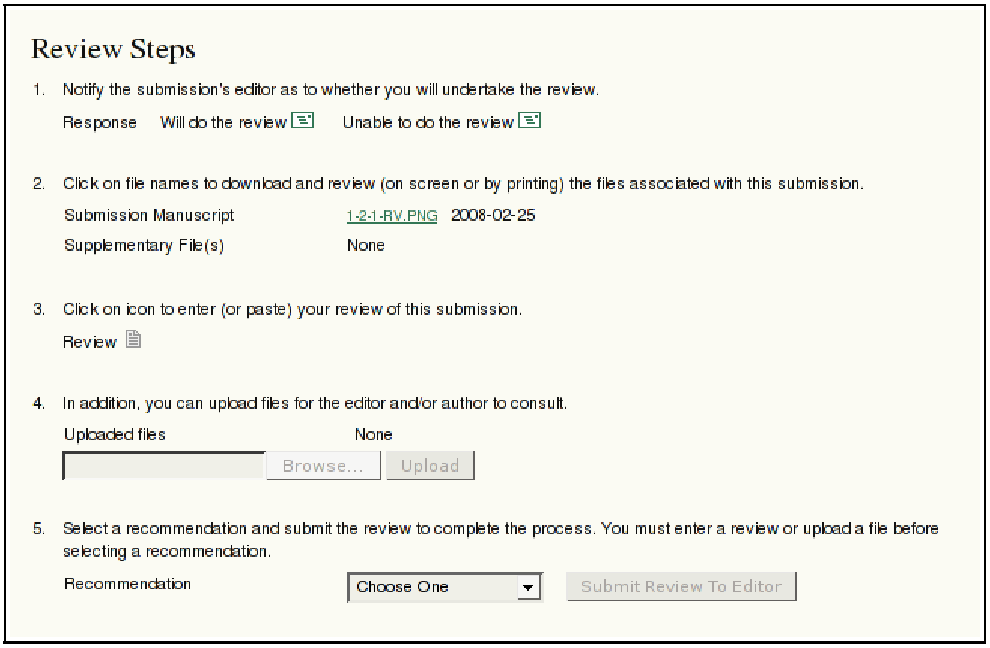
Once the Reviewers have been assigned, use the letter icon under the “Request” header to



compose a message requesting each Reviewer to consider undertaking the review:

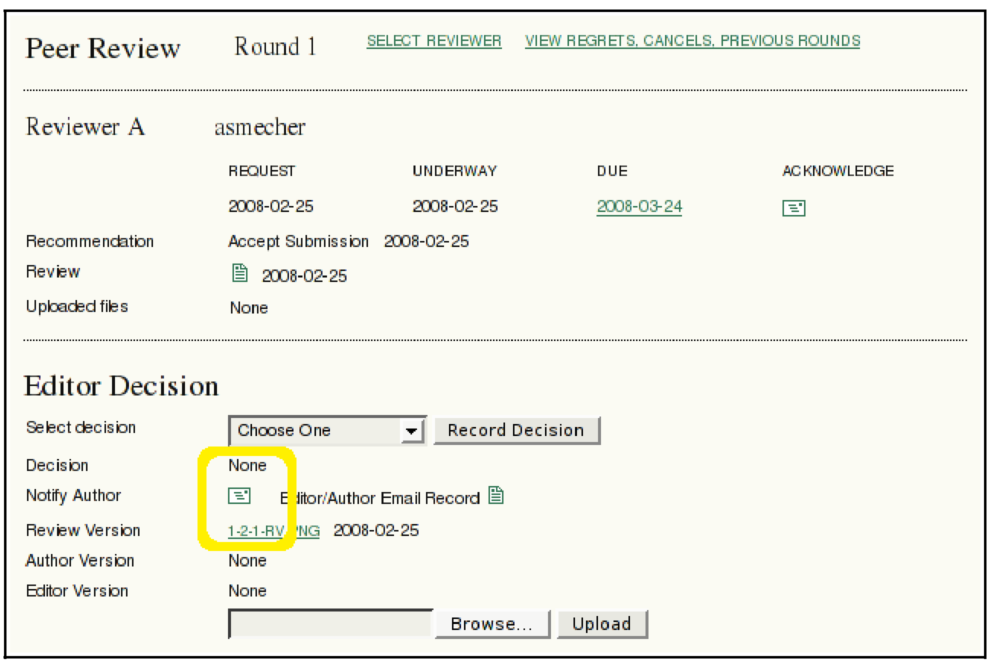
Once this email has been sent to the reviewers, they will be able to undertake or decline the reviews by clicking the URL included in the request message.

4. **Reviewer Performs Review.** Upon receipt of the reviewer request email, the reviewer will click on the included URL and be presented with a list of steps:



Following these steps one at a time, beginning with confirmation of whether or not the review request is accepted, will complete the review. Once the review has been completed, the Editor will be able to see the results on the submission's Review page.

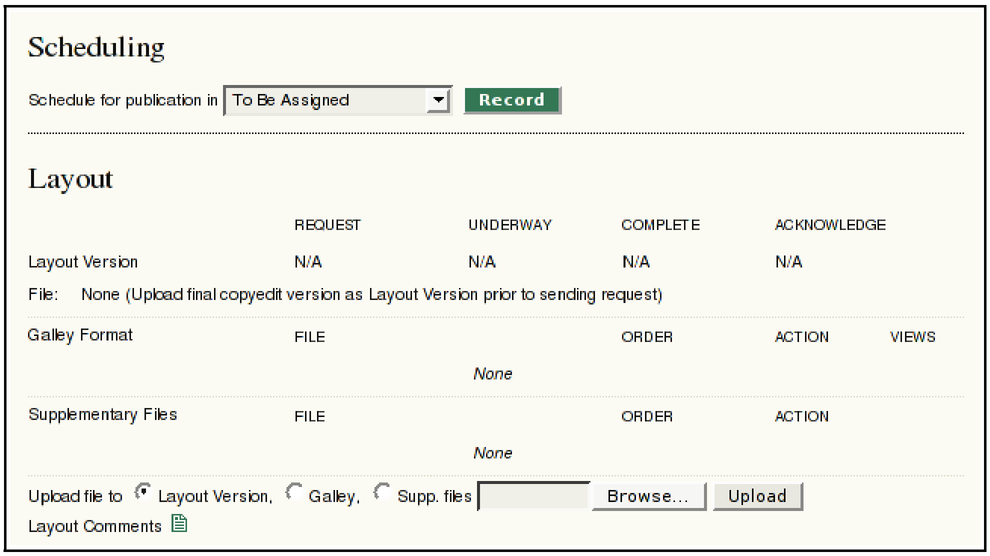
5. Editor Records Decision. Once all reviews have been completed, the Editor should record an editorial decision using the “Select decision” pulldown and “Record Decision” button. It is also important at this point to notify the Author of the results. Click on the letter icon beside “Notify Author”:



Once an "Accept" decision has been recorded, the Editor may choose a version of the submission under "Editor Decision" to Send to Copyediting. This will cause the submission to move from the "In Review" list to the "In Editing" list.

6. **Submission Editing.** Editing consists of Copyediting, Scheduling, Layout, and Proofreading processes. Depending on the journal configuration, these may be undertaken by the Editor, or they may be delegated to additional users.

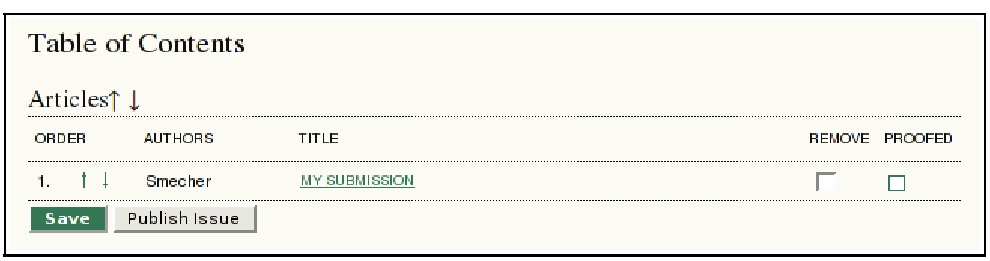
   Copyediting and Proofreading are optional processes and are not described here.



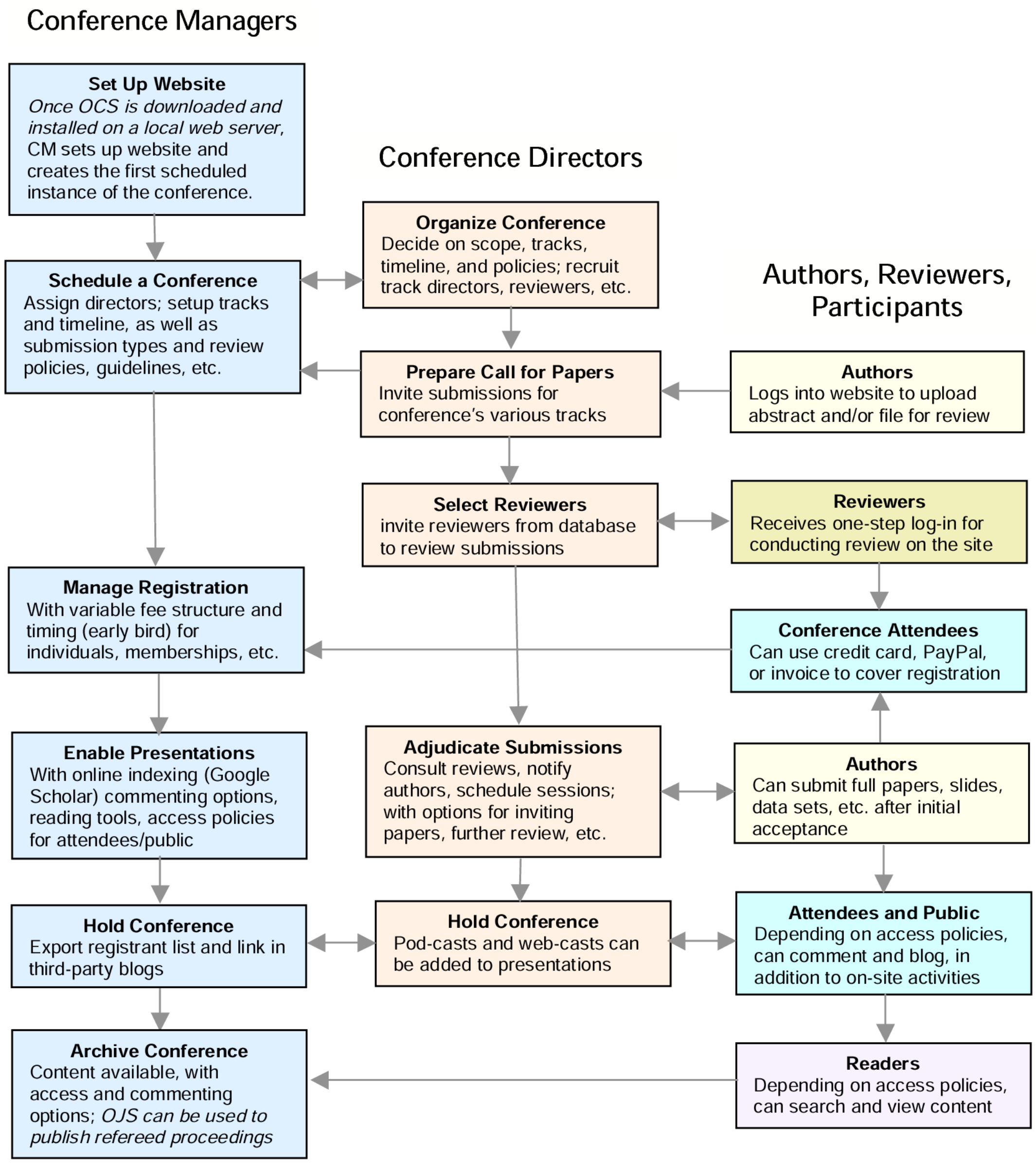
   Scheduling allows the Editor to choose which Issue of the journal the submission will be published in; Issues may be created by clicking "Create Issue" on the sidebar.

   Under Layout, ready-for-presentation versions of the submission (e.g. HTML and PDF) may be uploaded as Galleys.

7. **Publishing an Issue.** Once all appropriate submissions are scheduled against a particular Issue, the Editor can publish the Issue by clicking "Future Issues" (either from the sidebar or from the "Editor" link from "User Home") and choosing the issue. The Table of Contents page contains a "Publish Issue" button:



Open Conference Systems



Open Conference Systems supports several submission processes, including:

- Abstract-only
- Paper-only
- Abstract and paper with simultaneous submission and review
- Abstract then paper with separate review rounds

PKP PUBLIC KNOWLEDGE PROJECT

Creating Conferences (Open Conference Systems)

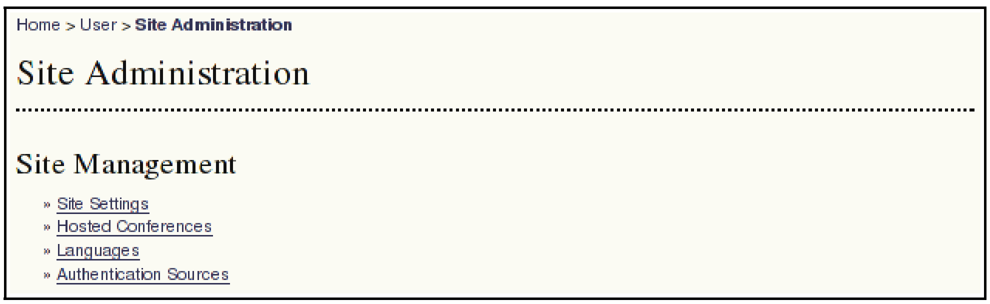
A single installation of OCS can host multiple conferences (e.g. “PKP Scholarly Publishing Conference” and “Special Topics in Computing”) and multiple instances of each conference (e.g. “PKP Scholarly Publishing Conference 2007” and “PKP Scholarly Publishing 2008”). The latter is called a “Scheduled Conference” in OCS. The Site Administrator is responsible for creating conferences, and the conference manager creates Scheduled Conferences.

Nearly all settings and options can be configured for each conference separately, so it is possible to host distinct conferences within the same installation without causing interference.

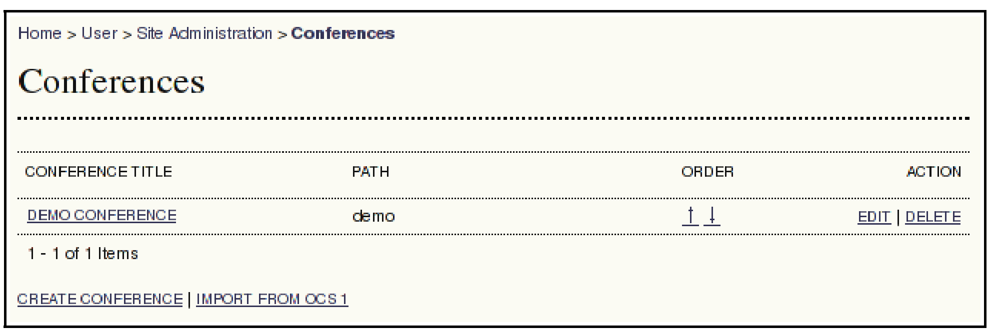
The user database is shared between conferences, however, so while it is possible for users to have different roles in different conferences, e.g. Conference Manager in one and Reader in another, it is not possible for users to have completely distinct profiles, e.g. names, biography statements, etc.)

Once logged in as Site Administrator, follow these steps to create a conference:

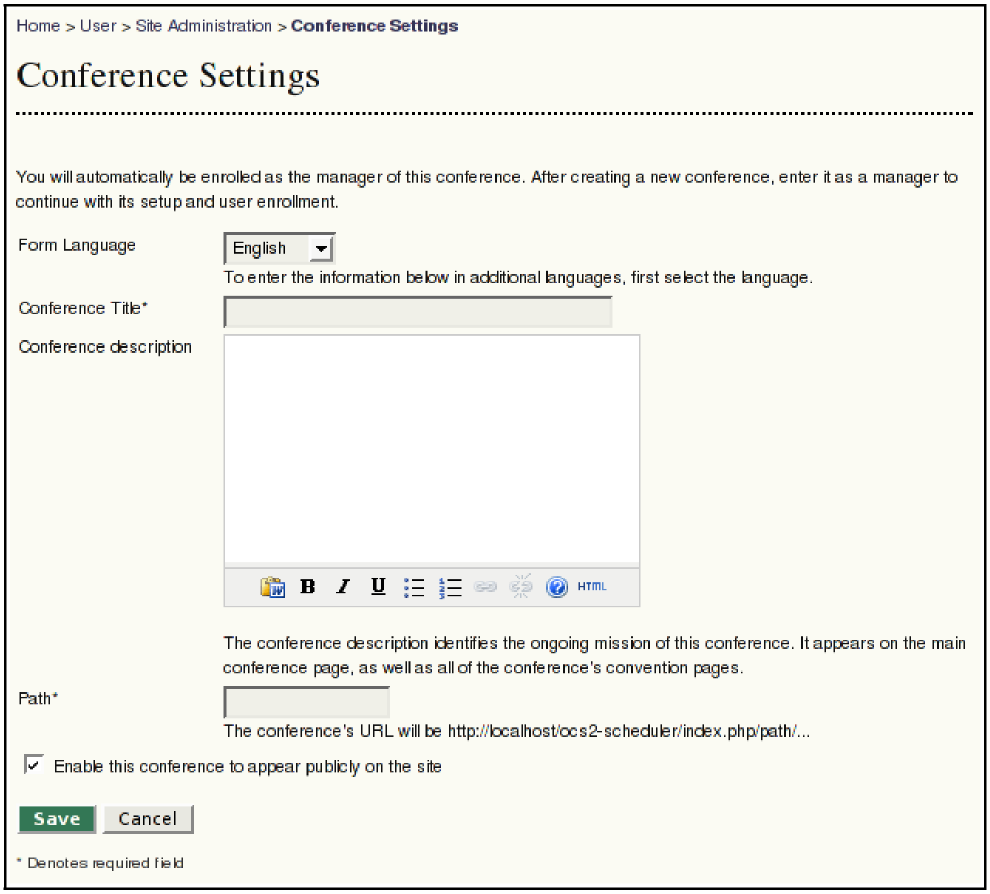
1. Find the “Hosted Journals” link in Site Administration:



2. Click on it to find a list of currently hosted conferences:

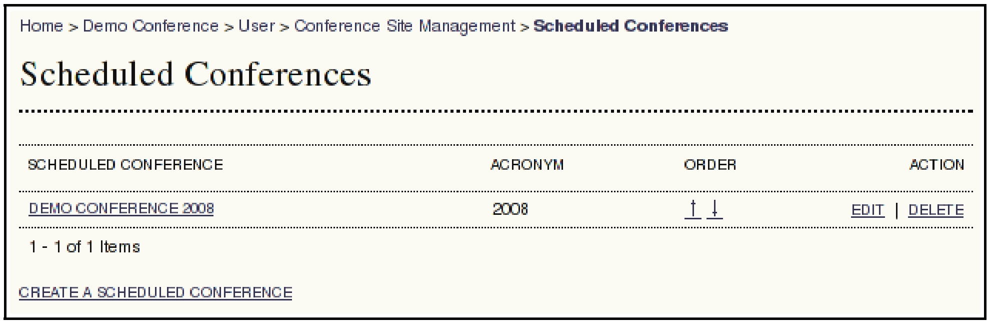


3. Follow the “Create Conference” link:

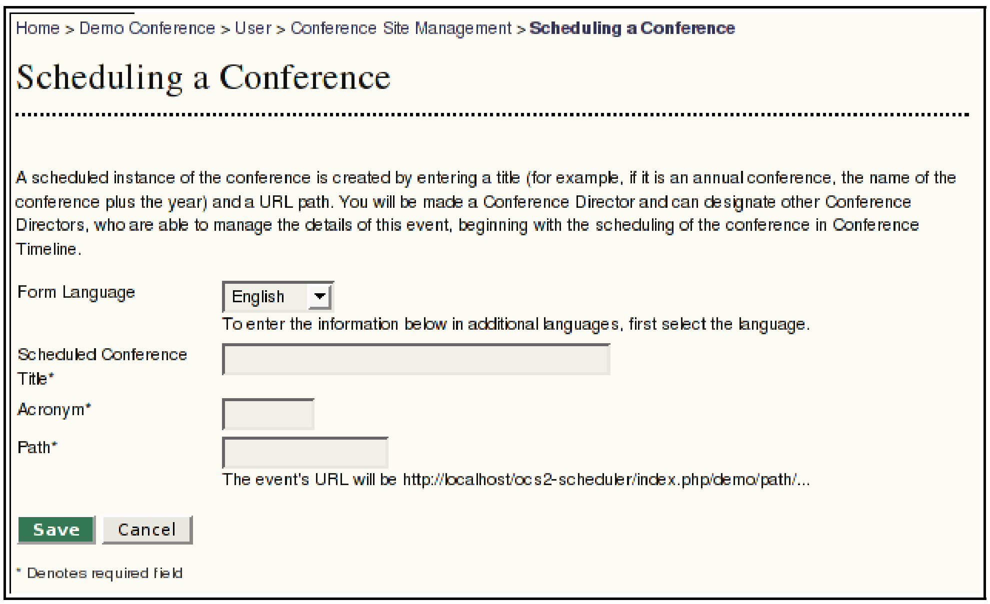


4. Enter Conference Management for the new conference:

5. Click "Scheduled Conferences" to enter the list of currently scheduled conferences for this conference:



6. Click "Create a Scheduled Conference" to enter the scheduled conference creation form:



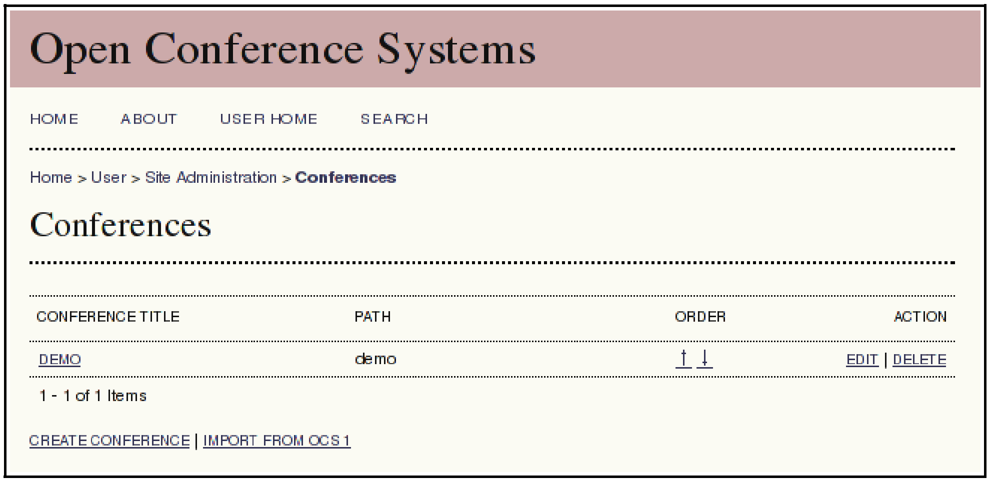
7. After the conference and scheduled conference have been created, it is strongly suggested that the Journal Manager at least review the "Website Management" pages for new Conferences, and "Setup" and "Conference Timeline" pages for new Scheduled Conferences. These can be found in "Conference Management", under "General Management" and "Current Conferences", respectively.

Steps to Publication (Open Conference Systems)

Requirements for Submissions

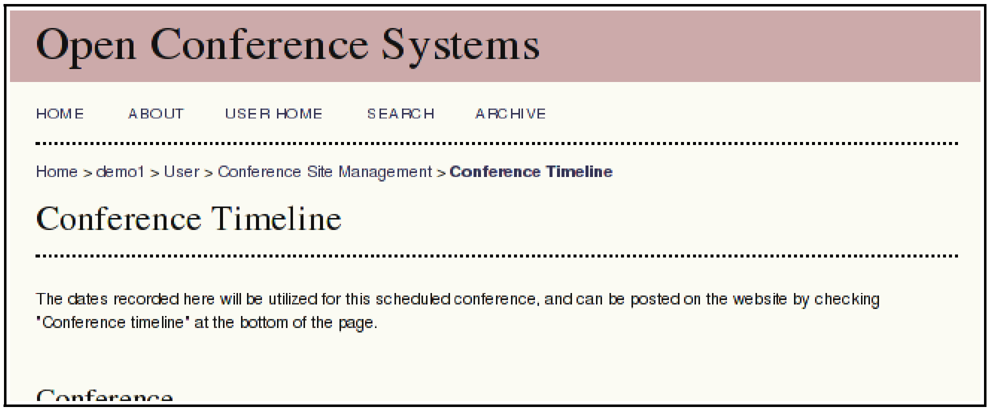
In order for submissions to be accepted, you will need a Conference and, within it, a Scheduled Conference for authors to submit to.

1. As Site Administrator, ensure that a Conference exists.



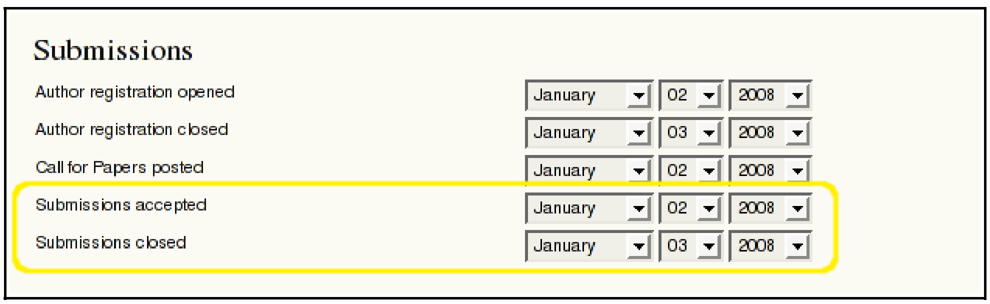
2. As Conference Manager, ensure that a Scheduled Conference exists.

3. As Journal Manager, find the Scheduled Conference's Timeline



page...

...and ensure that the "Submissions Accepted" and "Submissions Closed" dates are set appropriately. (Submissions will only be accepted on or after the Submissions Accepted date, and before the Submissions Closed date.)



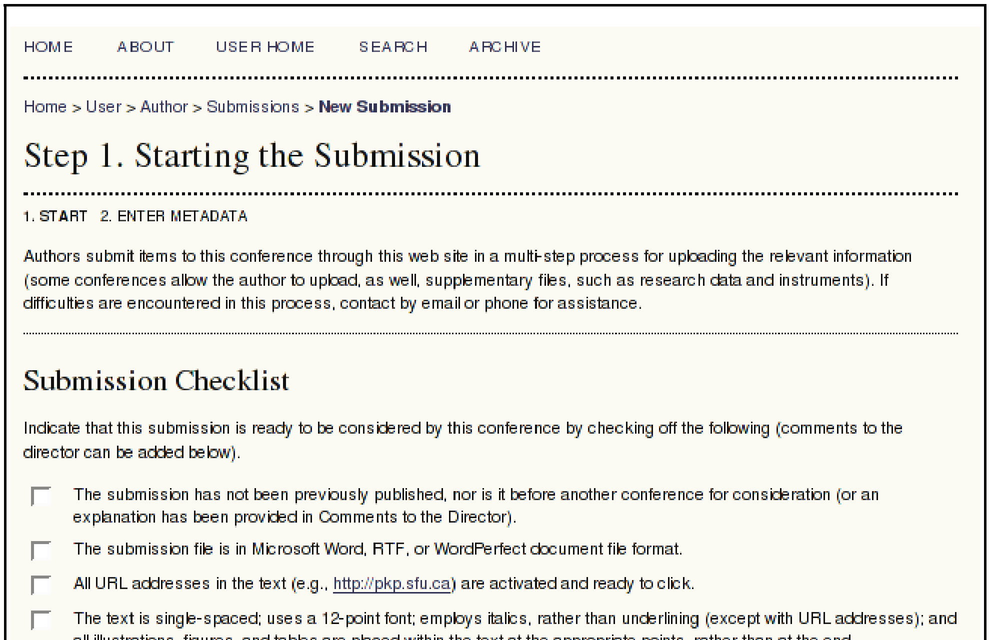
Submission to Publication

The following list describes the steps that a submission must undergo to reach publication. These steps describe the "Abstract and paper with simultaneous submission and review" process; the other processes (listed above) are similar.

1. **Author Submits Paper.** Potential Author follows a submission link, typically found on the conference Call for Papers page. (The Author may need to register with the site if they do not

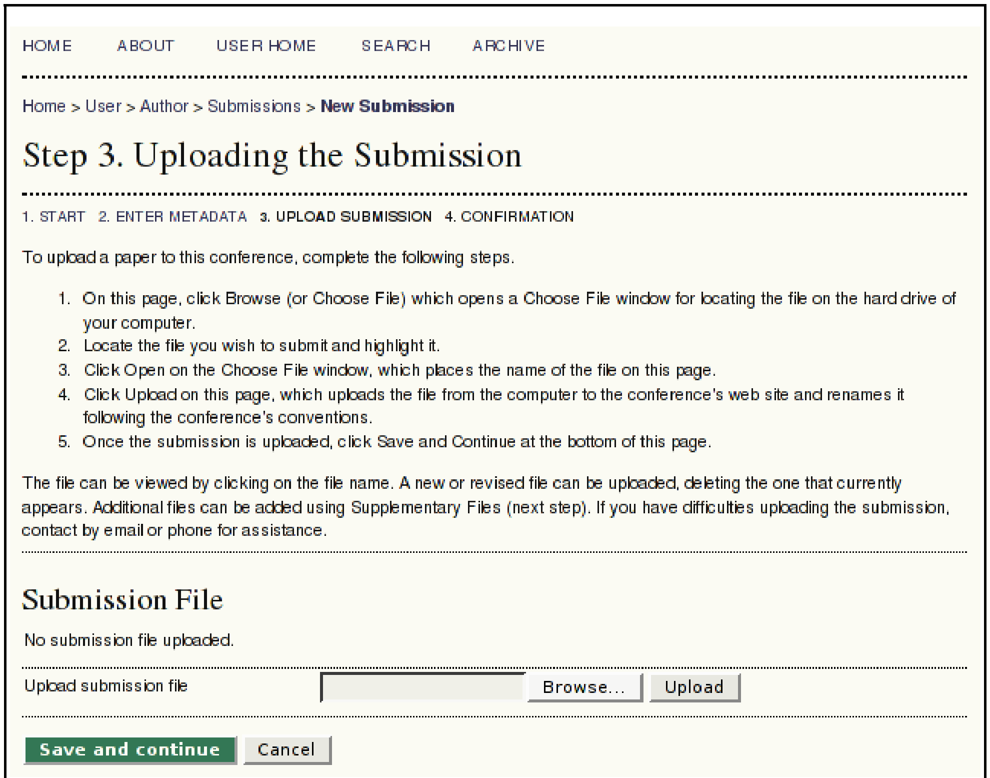
already have an account.)

2. Author completes the submission, including agreeing to terms of use, entering metadata, uploading the submission document, etc., depending on the submission process configured by the Journal Manager:



Submission process, step 1

Submission process, step 2



HOME ABOUT USER HOME SEARCH ARCHIVE

Home > User > Author > Submissions > **New Submission**

Step 3. Uploading the Submission

1. START 2. ENTER METADATA **3. UPLOAD SUBMISSION** 4. CONFIRMATION

To upload a paper to this conference, complete the following steps.

1. On this page, click Browse (or Choose File) which opens a Choose File window for locating the file on the hard drive of your computer.
2. Locate the file you wish to submit and highlight it.
3. Click Open on the Choose File window, which places the name of the file on this page.
4. Click Upload on this page, which uploads the file from the computer to the conference's web site and renames it following the conference's conventions.
5. Once the submission is uploaded, click Save and Continue at the bottom of this page.

The file can be viewed by clicking on the file name. A new or revised file can be uploaded, deleting the one that currently appears. Additional files can be added using Supplementary Files (next step). If you have difficulties uploading the submission, contact by email or phone for assistance.

Submission File

No submission file uploaded.

Upload submission file Browse... Upload

Save and continue Cancel

Submission process, step 3

HOME ABOUT USER HOME SEARCH ARCHIVE

Home > User > Author > Submissions > **New Submission**

Step 4. Confirming the Submission

1. START 2. ENTER METADATA 3. UPLOAD SUBMISSION **4. CONFIRMATION**

With the previous four steps of the submission process completed, click Finish Submission to submit your manuscript to demo. You will receive an acknowledgement by email and will be able to view your submission's progress through the editorial process by logging in to the conference web site. Thank you for submitting your work to demo.

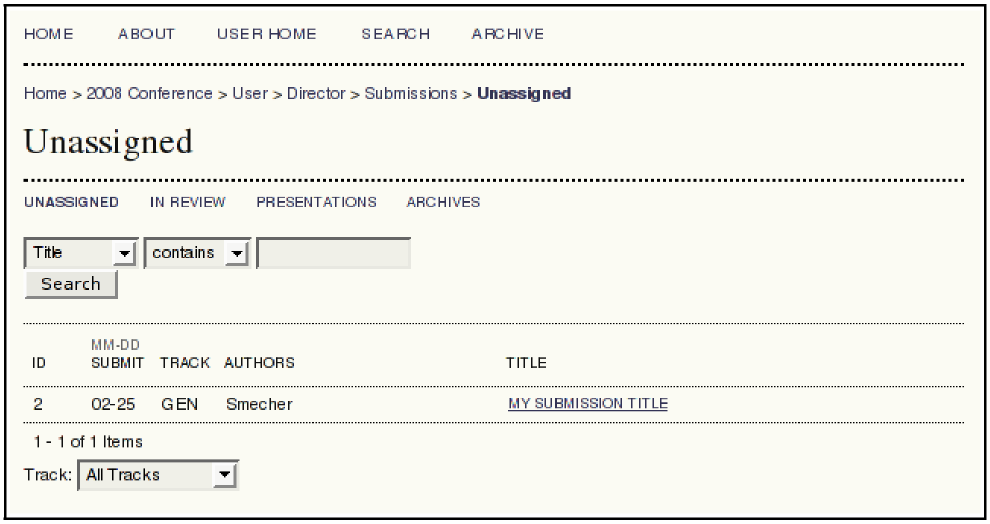
File Summary

ID	ORIGINAL FILE NAME	TYPE	FILE SIZE	DATE UPLOADED
1	789.PNG	Submission File	37KB	02-25

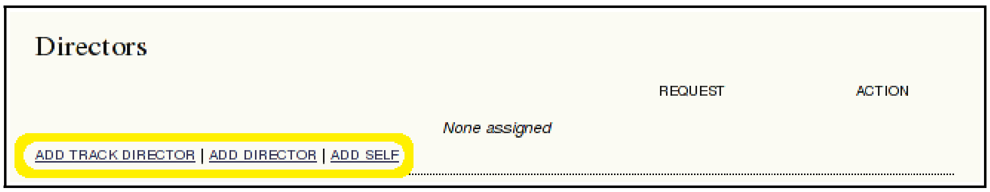
Finish Submission Cancel

Submission process, step 4

3. **Director Receives Submission.** Once the Author has completed these steps, the submission will appear in the Director's "Unassigned" or "In Review" submission list.



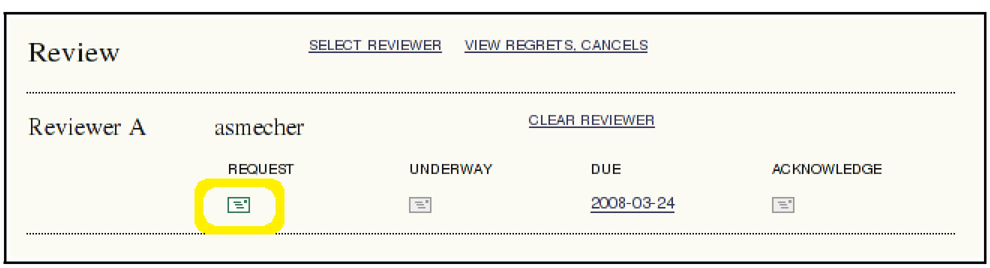
Click on the title to find the submission's Summary page. If it is not already assigned to at least one Director or Track Director, assign it using the Director's section of the Summary



page:

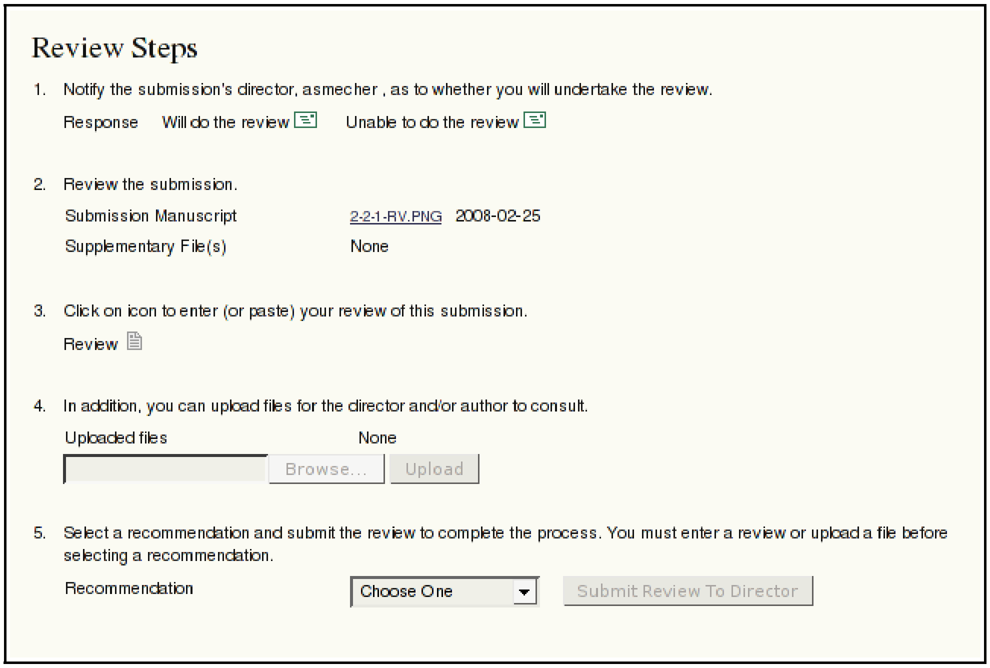
4. **Director Initiates Peer Review.** Go to the submission's Review page and assign reviewers as desired using the "Select Reviewer" link.

5. Once the reviewers have been selected, initiate the reviews by clicking the letter icon under "Request" and, if desired, customizing the letter to each Reviewer.



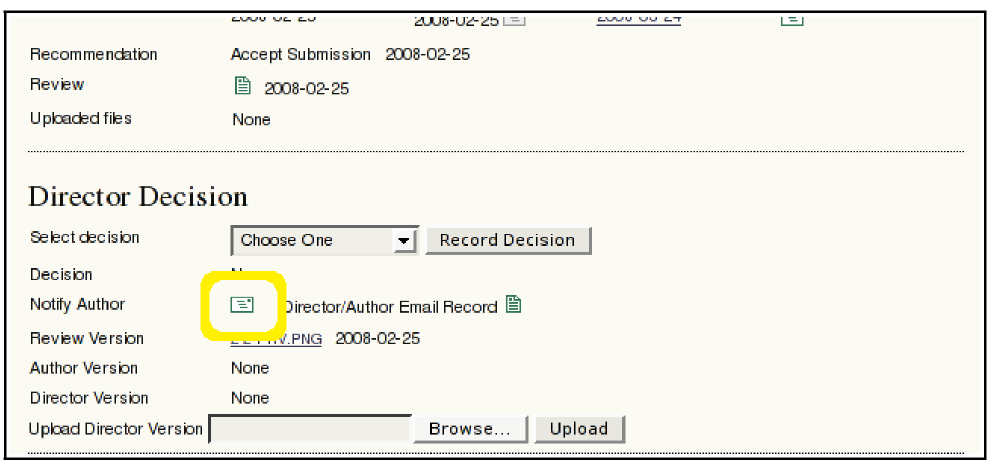
Once these messages have been sent, the Reviewers are able to begin the peer review process.

6. **Reviewer Performs Review.** The Reviewer will receive the email composed in Step 5, which will include a submission link. Following the link will show a list of necessary



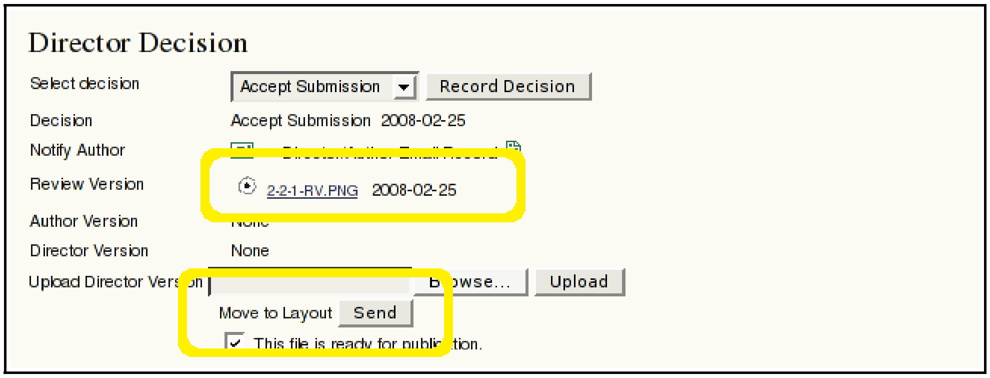
steps:
Following these steps one at a time, beginning with confirmation of whether or not the review request is accepted, will complete the review. Once the review has been completed, the Director will be able to see the results on the submission's Review page.

7. **Director Records Decision.** Once all reviews have been completed, the Director should record an editorial decision using the “Select decision” pulldown and “Record Decision” button. It is also important at this point to notify the Author of the results. Click on the letter icon beside “Notify Author”:



Compose a letter to the Author, using the “Import Peer Reviews” button to include the contents of any completed reviews that the Reviewer indicated are appropriate for the Author to read.

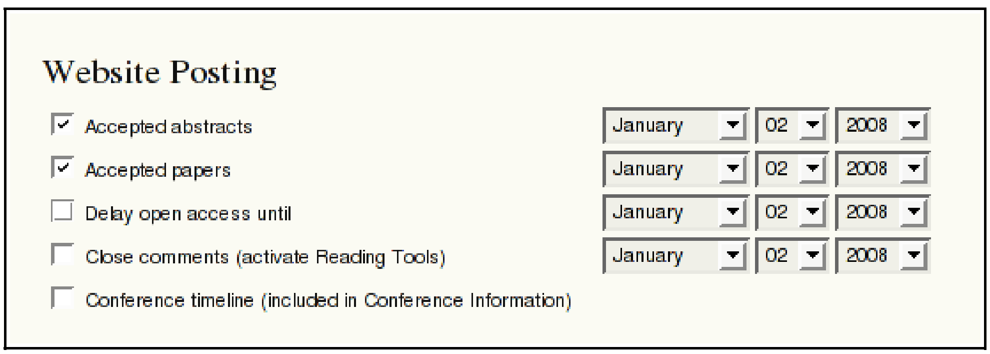
8. **Director Completes Submission.** Once the reviews are complete, the final version of an accepted paper must be prepared for the web. Under the “Director Decision” heading there may be several versions, depending on whether the Author has revised the submission and other considerations. Review the files, select the one to use, and click the “Send” button beside “Move to Layout”:



This will use an existing version of the submission for the paper that will be presented via the website. If you wish to generate your own layout versions, e.g. HTML or PDF, you may upload them as galleys under the “Layout” heading.

When the paper is finished, finalize the submission by clicking the “Complete” button.

9. **Making Papers Available.** Papers are published on the website according to the Conference Timeline, available under Scheduled Conference Management.



Registrations & Payment Processing

OCS supports payment processing for registrations. Particular payment services suppliers can be implemented using plugins, and OCS includes two as shipped: one supporting Paypal®, and another for manual payment processing. One of these must be configured by the Conference Manager before registrations will be allowed.

To configure registrations, follow these steps:

1. **Configure registration types.** As Conference Manager, find the “Registration” page. The three tabs along the top (“Registration”, “Registration Types”, and “Registration Policies”) allow the Conference Manager to control different aspects of the registration system; in particular, it is

important to make sure at least one Registration Type is configured, and that Registration Policies are set appropriately.

2. **Configure payments.** As Conference Manager, find the "Payment Methods" page. Choose the appropriate payment method (e.g. Paypal), and if any other fields appear on the form, configure them.
3. **Check the Conference Timeline.** For registration, it is important to ensure that the dates under "Website" are set appropriately.
4. **Double-check.** Log out of the Conference Manager account and check to see if registration is available. If using Paypal®, it is possible to use Paypal's development sandbox to test payments to ensure that they are being processed successfully. An alternative IPN URL is suggested on the OCS Paypal configuration form (step 2).

If using Paypal® to process payments, registrations will automatically be created in OCS. If using the manual payment method, it is up to the Conference Manager to provide sufficient information (e.g. forms that can be downloaded, printed, filled in, and mailed) to potential registrants.

Customizing Look & Feel

The appearance of each journal and conference can be changed independently. This can be done by choosing a theme, and/or uploading a custom Cascading Style Sheet (CSS) file. For OJS, this is done on Journal Setup, page 5; for OCS, this is done on Conference Setup, page 4.

When writing a stylesheet, be sure to override only the necessary styles. The standard stylesheet (`styles/common.css`) will be applied first, then potentially several additional stylesheets (e.g. for theme, sidebar content, etc.), then the custom stylesheet.

For example, to change the following stylesheet could be used to change the background colours of the page header and footer:

```
#header h1 {
	background-color: #069;
}

#footer {
	background-color: #069;
}
```

The stylesheets included with the themes, which can be found in `plugins/themes`, serve as typical examples.

Whenever possible, it is recommended to customize the look and feel of the application using CSS. This will present fewer challenges when upgrading to more recent releases, as modifications will not need to be reconciled against the new code. However, some kinds of customizations may not be possible using CSS alone; it may be necessary to alter the XHTML generated by OJS/OCS. This will involve modifying the Smarty templates included with the application.

Smarty templates, which have a .tpl suffix and reside in the "templates" subdirectory of the installation, resemble HTML documents with special "Smarty" tags that perform processing before the document is sent to the user's browser. Some common Smarty tags are:

- `{translate key="some.key.here"}` inserts a piece of language-specific text identified by "some.key.here" from an XML file (e.g. locale/en_US/locale.xml for the main English locale file)
- `{include file="common/header.tpl"}` includes the main `header.tpl` template (which is used for most pages in OJS/OCS)
- `{iterate from=conferences item=conference}` ... `{/iterate}` loops through all conferences supplied by the "conferences" iterator (this is used, for example, in OCS in `templates/admin/conferences.tpl` to provide the Site Administrator with a list of all conferences)

See http://php.smarty.net for further Smarty documentation.

Template files are arranged logically into subdirectories. For example, the main "About" template is located in `templates/about/index.tpl`.

If altering Smarty templates for layout, the Smarty tags themselves can typically be left untouched; modifications should only be necessary to the surrounding HTML tags. The most commonly modified templates are `templates/common/header.tpl` and `templates/common/footer.tpl`, which define the overall structure (e.g. navigation links etc.) of nearly every page generated by OJS or OCS.

Server Administrator

The installation process is documented in `docs/README`.

Minimum Server Requirements

- PHP >= 4.2.x (including PHP 5.x); Microsoft IIS requires PHP 5.x
- MySQL >= 3.23.23 (including MySQL 4.x and 5.x) or PostgreSQL >= 7.1 (including PostgreSQL 8.x)
- Apache >= 1.3.2x or >= 2.0.4x or Microsoft IIS 6
- Operating system: Any OS that supports the above software, including Linux, BSD, Solaris, Mac OS X, Windows

Recommended Server Configuration

- PHP 5.x with iconv, mbstring, libgd, and libfreetype support
- MySQL 5.x with data storage and connection using UTF8
- *NIX-like OS (Linux, BSD, Mac OS X)
- Apache web server configured with PHP via FastCGI

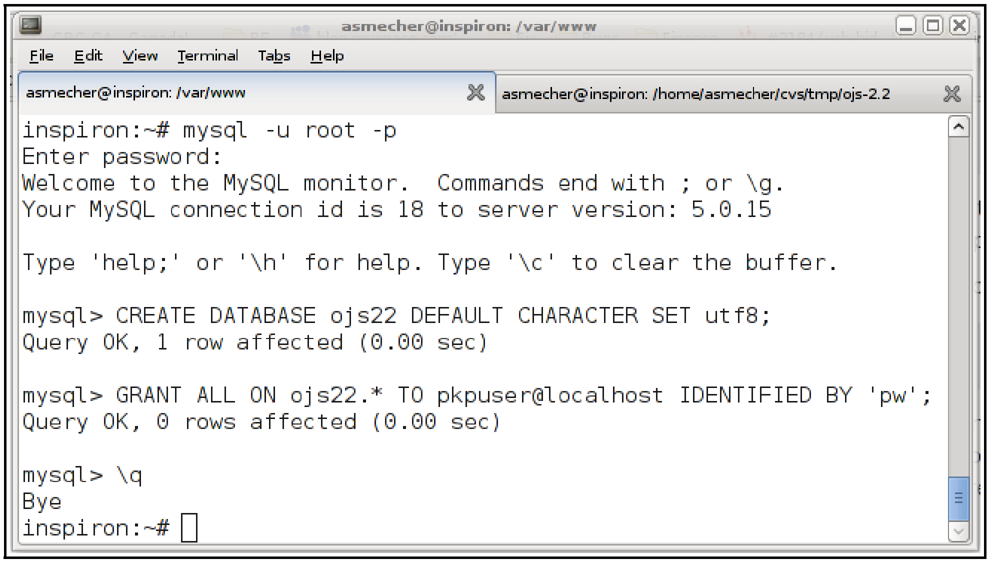
Components of an Installation

- Installation path containing OJS or OCS (typically 10-20MB)
- Files path, as configured in `config.inc.php` using the `files_dir` directive; size varies depending on the documents being managed (e.g. file format, design complexity, number of rounds of review, etc.)
- MySQL database, as configured in `config.inc.php` in the `[database]` section; size varies from tens of megabytes for small journals to hundreds for large journals or large collections
- OJS and OCS contain several third-party open source libraries:
  - ADODB (http://adodb.sourceforge.net)
  - Smarty (http://smarty.php.net)
  - TinyMCE (http://www.moxiecode.com)

Installation

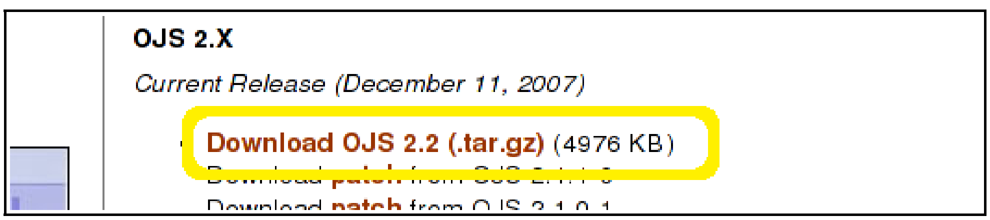
Installation instructions will differ depending on the specifics of the system being used. These instructions apply to Linux/Unix systems but will be substantially similar for Windows. The system requirements listed above must be met before beginning the installation process. The following instructions are for OJS, but will work for OCS.

1. Create a new MySQL database for the OJS installation to use.

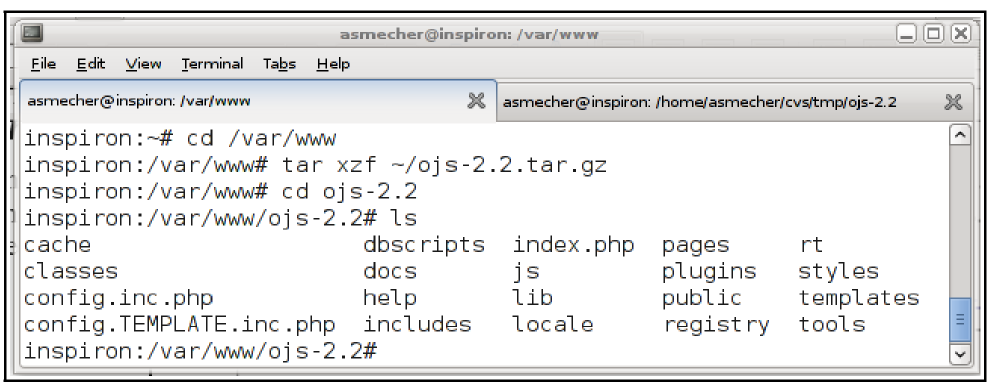


   In this example, the database is called `ojs22`, the database user is called `pkpuser`, and the password is `pw`. You will need these three items later.

2. Download the OJS/OCS installation package from the PKP website (http://pkp.sfu.ca).

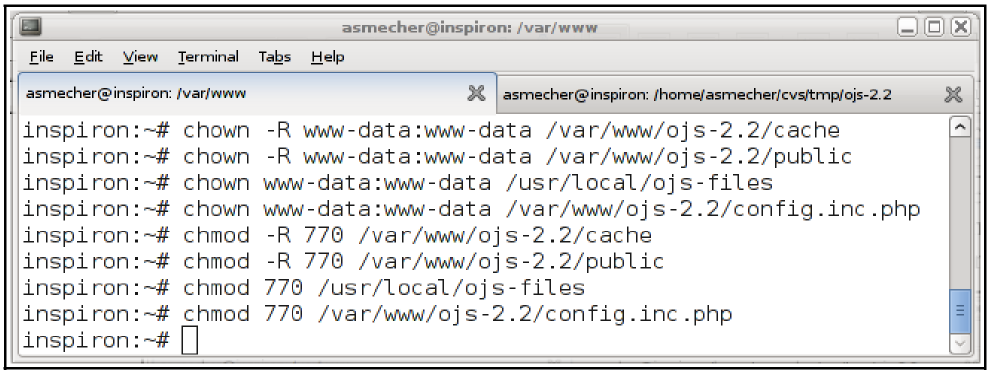


3. Unpack the archive into the web server's root directory. (This will create a directory called e.g. `ojs-2.2` containing the installation files.)

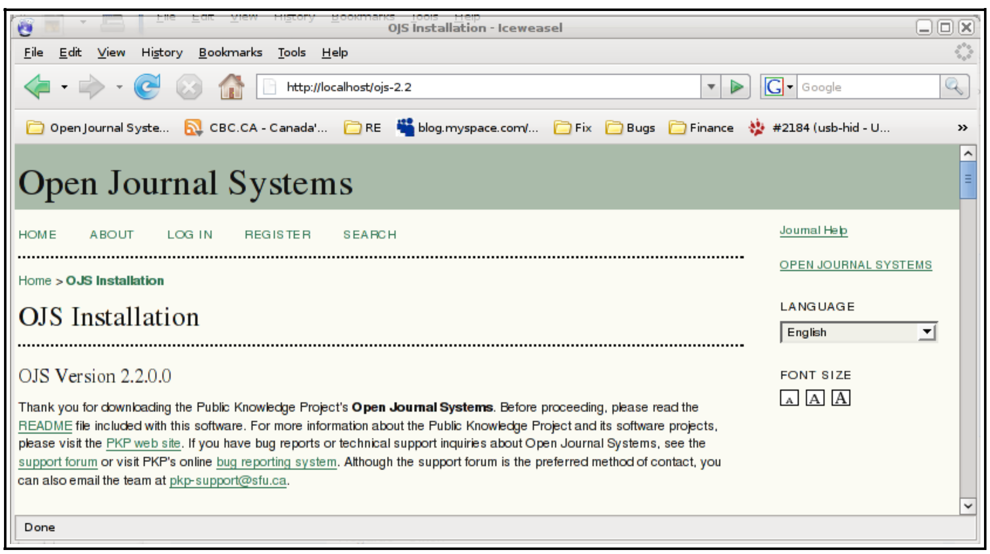


4. Make a directory for OJS to store its files in. This should *not* be a subdirectory of the web root, or it may be possible for private files to be accessed via the web server.

5. Grant file permissions so that the web server can administer the `public` and `cache` subdirectories of the OJS installation path, the files path created in step 4, and the `config.inc.php` configuration file. The specifics of this step will depend on your web server configuration, i.e. whether PHP scripts run SetUID.



6. Point your web browser to the OJS installation directory.

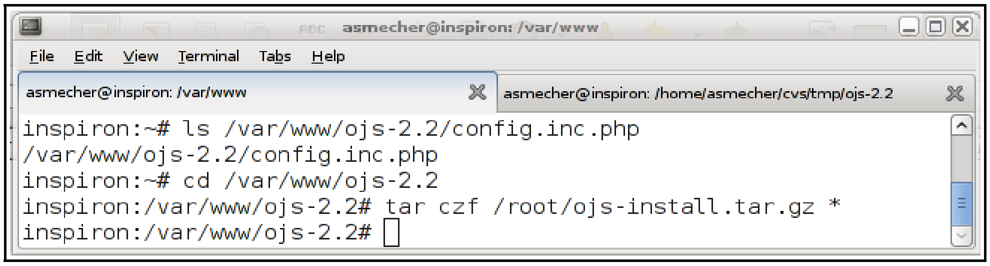


Fill in the form (including the database connection information you used in step 1, and under “File Settings”, the location of the files path you created in step 4) and click the “Install” button at the bottom.

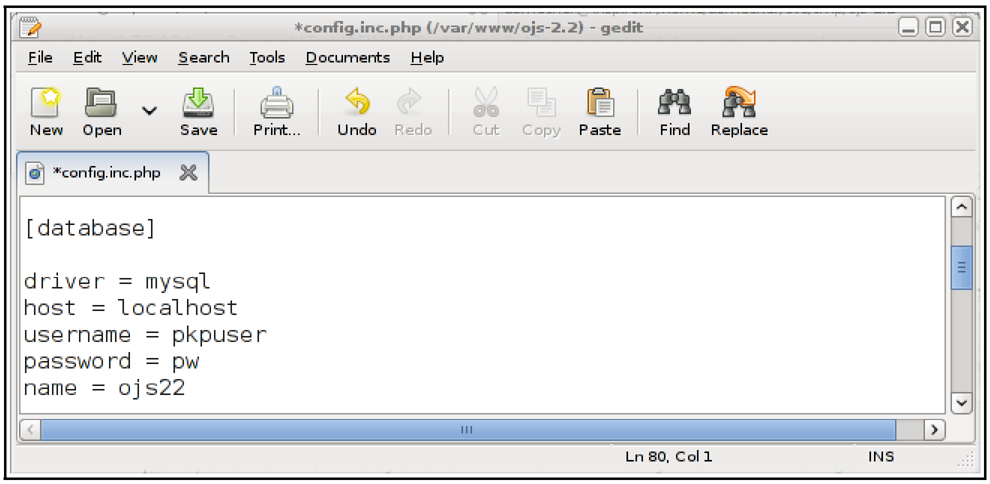
Backups

A full back up of an OJS or OCS installation should include three components:

- The OJS installation itself. To find it, look for the `config.inc.php` configuration file; this will be in the installation directory. Back up all files in this directory and all subdirectories. Assuming OJS is installed in `/var/www/ojs-2.2` and the backup should be placed in `/root`:



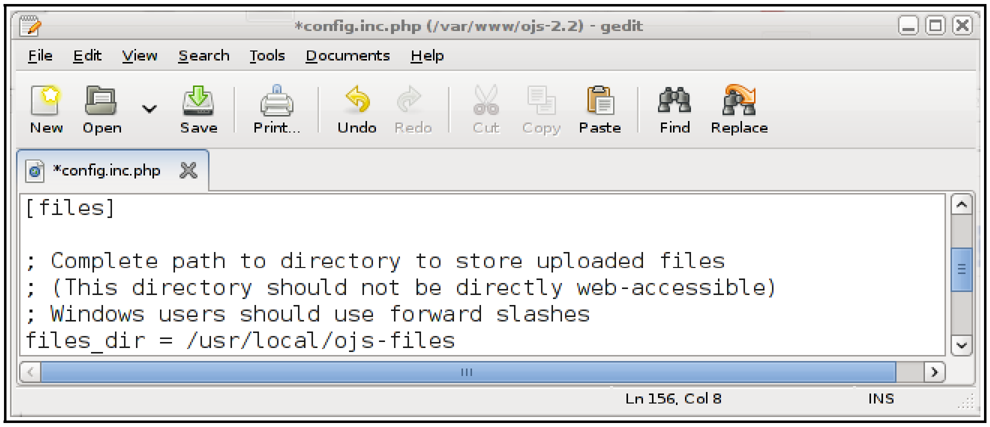
- The MySQL database. You will need to know the database name, username, and password; these can be found in the `config.inc.php` configuration file:



  Back up the database using the `mysqldump` tool, e.g. if the backup should be placed in `/root`:

  ```
  mysqldump -u pkpuser -p ojs22 | gzip -9 > /root/ojs-database.sql.gz
  ```

- The files path. To find out where this is, look in the `config.inc.php` configuration file:

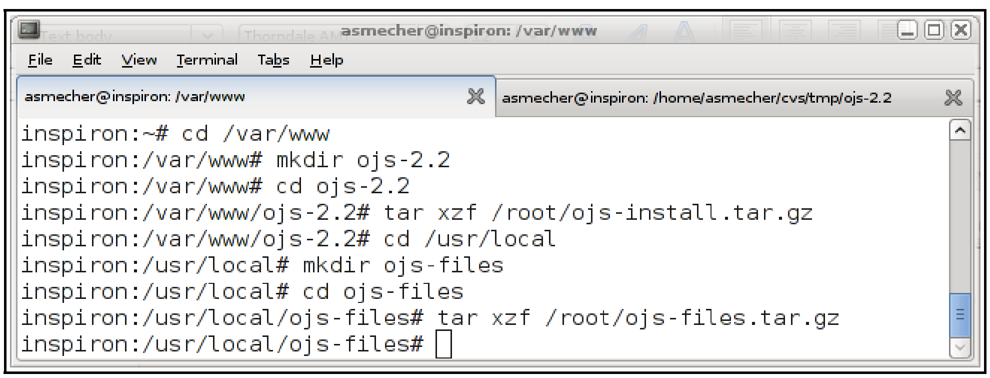


Back up this directory and all of its contents, e.g. if the backup should be placed in `/root`:

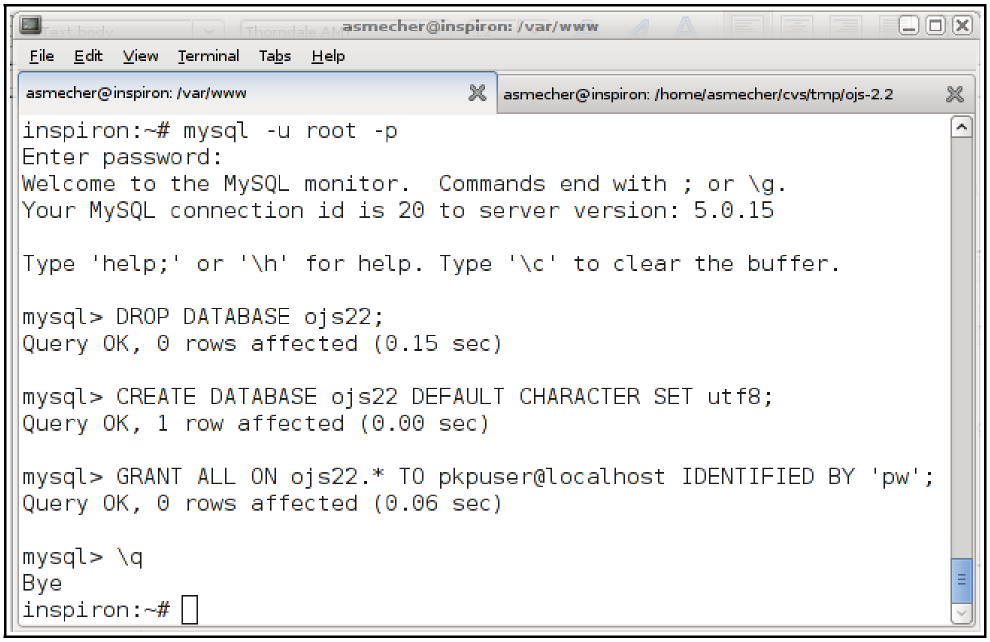
When these steps have been completed, you will have created three backup files. All three of these are required to restore a copy of OJS from backup.

Restoring

To restore from backup, unpack the files and installation backups into their appropriate places:



Re-create the MySQL database (dropping it first if necessary):



```
inspiron:~# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 20 to server version: 5.0.15

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> DROP DATABASE ojs22;
Query OK, 0 rows affected (0.15 sec)

mysql> CREATE DATABASE ojs22 DEFAULT CHARACTER SET utf8;
Query OK, 1 row affected (0.00 sec)

mysql> GRANT ALL ON ojs22.* TO pkpuser@localhost IDENTIFIED BY 'pw';
Query OK, 0 rows affected (0.06 sec)

mysql> \q
Bye
inspiron:~#
```


Load the database dump into the newly created database:

```
zcat /root/ojs-database.sql.gz | mysql -u pkpuser -p ojs22
```

Upgrades

Comprehensive upgrade instructions can be found in `docs/UPGRADE`. **As with any major administration task, it is strongly suggested that you back up your complete installation before beginning.** Upgrading is a two-step process:

1. The code must first be upgraded. This will involve downloading a new copy of OJS or OCS, or downloading and installing a patch file.
2. The database must be upgraded. This can be done using the `tools/upgrade.php` script.

See `docs/UPGRADE` for complete instructions.

If the installation has been customized in any way, it will be necessary to ensure that the customizations are properly migrated to the new version:

- Use the standard `diff` and `patch` tools. To generate a patch file describing the modifications you have made to an installation, generate a diff file against a stock version of that installation. For example, if you are upgrading from OJS 2.1.1 to OJS 2.2:
  `diff -u /path/stock-ojs-2.1.1 /path/modified-ojs-2.1.1-install > mods.diff`
  This will generate a file called `mods.diff`. Review this file; these changes may need to be transferred into the upgraded installation.
- Ensure that any schema changes you make are properly reflected in the XML database schema descriptor(s) in `dbscripts/xml` (e.g. `dbscripts/xml/ojs_schema.xml`). During upgrades, these schema files are applied and any added or modified columns may be reverted.

Notes on Administration

- If your server is configured to invoke PHP scripts via CGI or FastCGI, it is best to configure OJS and OCS to run in a setUID environment. Servers running PHP scripts via `mod_php` are often not secure, as all PHP scripts on the server typically run with a single user account.
- The files path configured in `config.inc.php` using the `files_dir` directive should **not** be accessible directly via the web server, i.e. it should not be a subdirectory of your installation path.
- PHP Safe Mode is "broken", in that directories created by PHP scripts are usually created with default file permissions such that that they cannot be used by those same scripts. It is possible via `php.ini` to relax the strict ownership check to a `gid` check, which may help if you are forced to use a Safe Mode environment.
- When moving an OJS or OCS installation to another server or otherwise changing URLs, it may be necessary to manually correct any URLs that are specified in the Journal, Conference, or Scheduled Conference setup pages. Double-check the `config.inc.php` file to ensure that the `base_url` setting is correct. OJS should otherwise be able to detect the appropriate URL to use.
- There is a Frequently Asked Questions document that contains additional information; see `docs/FAQ` in your installation for further details.
- The support forum (http://pkp.sfu.ca/support/forum) contains a large knowledge base of common problems and solutions.

Translations

Translations in OJS/OCS are maintained in a set of XML files. Each translation is given a standard symbolic name identifying the language and locale for which it is intended; for example, `en_US` indicates the English language as spoken in the United States.

All XML files should be saved using the UTF-8 encoding.

There is a list of languages in `registry/locales.xml`; all languages must be included there before they will appear e.g. in the Site Administrator or Journal Manager's "Languages" pages.

When working with XML, it is important to have a good text or XML editor that supports UTF-8, and a validator. Oxygen and XMLSpy are popular examples for Windows; on a *NIX platform, `vi` and `xmllint` form an excellent combination. To prevent typos from causing confusing problems, validate any modified XML files before attempting to use them with OJS.

For each language, there are a number of files that need to be translated. When creating a new translation, begin with the English (`en_US`) version of the file and translate all textual content into the new language.

Locale Files

These XML files map from a symbolic name for each piece of text (e.g. `form.errorsOccurred`, which explains to a user that a form could not be saved) to a translated piece of text for a given language (e.g. `xx_XX` in the list below).

- `locale/xx_XX/locale.xml`: This is the main language file, containing most of the translated text for the user interface.
- `plugins/*/*/locale/xx_XX/locale.xml` (one for each plugin category and plugin, e.g. `plugins/importexport/native/locale/en_US/locale.xml`): These are plugin-specific locale files, which provide translated text for each plugin.

Additional Important Language Files

In addition to the locale files, there are several other aspects of OJS and OCS that require translated content. These are:

- `dbscripts/xml/data/locale/xx_XX/email_templates_data.xml`: This file contains the set of email templates that are included with the system. (Note: This file will be moved into `locale/xx_XX` in a future release.)
- `registry/locale/xx_XX/countries.xml`: This is a specially-formatted version of the ISO standard list of countries (ISO 3166). (Note: This file will be moved into `locale/xx_XX` in a future release.)
- `locale/xx_XX/currencies.xml`: This is a specially-formatted version of the ISO standard list of currencies (ISO 4217).

Optional Components of a Translation

The above two sets of files are required for a translation to be considered complete. There are a few additional elements that are considered optional:

- `help/xx_XX`: This directory, and numerous subdirectories, contain the help content for the online help system included in OJS and OCS.
- `rt/xx_XX`: This directory contains the set of Reading Tools supported for each language.
- `locale/xx_XX/edprocesslarge.png` and `locale/xx_XX/edprocess.png`: These images (available in "Help" and "About This Publishing System") describe the editorial process for the application.

Creating a Translation

To create a new translation:

1. Find the symbolic name for the new translation. These symbolic names are standardized and must be used consistently when translating. `xx_XX` is used here as an example.
2. Add the new locale to the system locale list, `registry/locales.xml`.
3. Copy the main English (`en_US`) locale file (`locale/en_US/locale.xml`) into the

appropriate location for the new translation (e.g. `locale/xx_XX/locale.xml`) and translate its contents.

4. Copy the `en_US` (`dbscripts/xml/data/locale/en_US/email_templates_data.xml`) email file into the appropriate location for the new translation and translate its contents.
5. Validate the XML files you created or modified in steps 2, 3, and 4.
6. Log in as Site Administrator and install the new language from the Languages page.
7. Test the new language as Journal Manager (or Conference Manager in OCS) by going to that role's Languages page and enabling the new language. Choose the new language from the sidebar.
8. For each of the remaining locale files and additional important language files listed above, create a translated version, install it into the appropriate location, validate it, and test it.
9. Please consider contacting the PKP team at pkp-support@sfu.ca so that we can distribute the translation with future releases of OJS/OCS, and so that we can help maintain the translation for future releases.

Translation Tools

Comprehensive tools are available in the form of a plugin to help check and maintain translations. These are not recommended for creating new translations, but are extremely useful for updating translations to conform to subsequent software releases.

Debugging Translations

1. When encountering difficulties with a translation, it is strongly recommended that the translator validate all the XML files involved (using a desktop XML editor's validation features, or an external tool such as `xmllint`).
2. If locale file changes are not being reflected in the user interface, check the file permissions on the `cache` directory and all of its contents. The web server must be able to administer (create, modify, delete) all contents of the `cache` directory and its subdirectories.
3. If strings such as `##symbolic.name.here##` are appearing in the user interface, it usually means that a locale string (in this case `symbolic.name.here`) is missing from a locale file. Check which English file the string is defined in, and add it to the respective file in your translation.
4. If problems are encountered with diacritical marks (accented characters) or symbols, ensure that your text editor is configured to save files using the standard `UTF-8` character encoding. (External tools are available to convert files between different encodings, such as `iconv`.)

Developer

There is a comprehensive technical reference available on the PKP website.

Additional References & Resources

- PKP Bugzilla: http://pkp.sfu.ca/bugzilla
- PKP CVSWeb: http://pkp.sfu.ca/cvs
- Smarty reference: http://smarty.php.net
- ADODB reference: http://adodb.sourceforge.net
- PHP reference: http://php.net
- MySQL reference: http://dev.mysql.com/doc/refman/5.0/en/

File Suffixes

The following file suffixes are commonly used in an OJS/OCS installation:

- `php`: These files contain the PHP code that comprises most of the system.
- `tpl`: These files are Smarty templates, and define the HTML that is generated by the system to send to the user's web browser. These resemble HTML, but contain Smarty tags (e.g. `{translate key="some.key.here"}`) which are executed on the server before the result is sent to the user.
- `xml`: OJS/OCS contain many XML data files that control the operation of the system. Most of these are locale files, which contain language-specific text (e.g. `locale/en_US/locale.xml`); however, there are many others, such as `dbscripts/xml/ojs_schema.xml` (for OJS) or `dbscripts/xml/ocs_schema.xml` (for OCS), which describe the database structure. All XML files should be UTF-8 encoded.

Useful Commands

These commands are intended for a Linux environment or similar (e.g. MacOSX, *BSD). On a Windows platform, use the "Find" tool, limiting file extensions to `php` or `tpl`, to achieve the same.

- To find a piece of code referencing a particular function (e.g. Request::getRequestedOp):
  ```
  fgrep -l "Request::getRequestedOp" `find . -name \*.php`
  ```
  (Note back-ticks)
- To find where a localization key appears in the template files (e.g. common.saveAndContinue):
  ```
  fgrep -l "common.saveAndContinue" `find . -name \*.tpl`
  ```
  (Note back-ticks)

Plugin Categories

Each of the plugin categories supported by OCS and OJS is listed below. A plugin should be installed in the subdirectory of the "plugins" directory indicated in the list below, and should extend the parent class listed in brackets.

- `auth` (`classes/plugins/AuthPlugin.inc.php`): Additional means of authenticating user accounts (e.g. via LDAP) can be implemented using auth plugins.
- `blocks` (`classes/plugins/BlockPlugin.inc.php`): Pieces of content that appear on the sidebar (e.g. the "user" block, which provides login and logout links among others) are implemented using sidebar block plugins.
- `citationFormats` (`classes/plugins/CitationPlugin.inc.php`): Each citation format that is available via the Reading Tools sidebar is implemented as a citation format plugin.
- `gateways` (`classes/plugins/GatewayPlugin.inc.php`): Gateway plugins expose functionality to external services at a specifically-crafted URL. This URL takes the form:
  `http://`*`url/to/installation`*`/index.php/`*`journalPath`*`/gateway/plugin/`*`PluginName`*
  ...where `url/to/installation`, `journalPath`, and `PluginName` are replaced with specific information about the installation, journal, and plugin being queried. (OCS is similar.)
- `generic` (`classes/plugin/GenericPlugin.inc.php`): Generic plugins can be used to override various parts of the system that are not better suited to other types of plugins. Generic plugins are always loaded and are loaded early in the process of handling a request. Hooks are generally intended for use with generic plugins.
- `importexport` (`classes/plugin/ImportExportPlugin.inc.php`): The Journal or Conference Manager can invoke import/export plugins to manually extract data from or insert data into the system. These are typically invoked via the Manager's user interface, but can also be called using the `tools/importexport.php` command-line script.
- `paymethod` (`classes/plugin/PaymethodPlugin.inc.php`): Payment methods and services (e.g. PayPal) can be implemented using paymethod plugins.
- `themes` (`classes/plugin/ThemePlugin.inc.php`): Theme plugins can be used to customize the appearance of the system, typically by applying CSS (Cascading Style Sheets), but potentially by performing other tasks like filtering generated HTML (e.g. using Smarty postfilters) or overriding templates.

Locating Handling Code

To locate a piece of code, begin with the URL to the request, e.g. for OJS:

    http://localhost/ojs2/index.php/demo/manager/setup/5

The "index.php" part of the URL indicates the main wrapper for the system. In OJS, the next part ("demo") indicates the journal path. In OCS, there are instead two parts (the conference path and scheduled conference path).

The key parts are "`manager`" and "`setup`", which are the page name and operation, respectively. Find handling code by looking in `pages/manager/ManagerHandler.inc.php` for the "`setup`" function. (Any further parts to the URL, in this case "5", are passed to the function as an array parameter.)

Debugging

An FAQ document is included with each application in `docs/FAQ`. Depending on the server configuration, PHP error messages may either be displayed to the remote browser (convenient for development but not recommended for live servers) or sent to the log file; to ensure that they are sent to the browser, insert the following just below the `<?php` line of the main `index.php` wrapper:

```
ini_set('display_errors', E_ALL);
```

If you encounter a blank page while developing and are unable to find a PHP error message, it is likely due to a syntax error in your PHP code. Back the changes out gradually until the page starts working again, then re-introduce them step by step until the error is isolated.

Debugging Performance Issues

If you are experiencing poor performance, it is necessary first to determine which of the two most likely causes is at play: PHP/Apache or MySQL. To do this, use a server tool such as `top` to determine whether requests are resulting in a heavy load on MySQL or on PHP/Apache.

- If Apache or PHP are experiencing heavy loads, you may need to look into a few aspects of the server's configuration:
    - Are you running a PHP optimizer? Zend Optimizer can help improve performance considerably, but ensure you are using a new version, as older versions have bugs that affect OJS/OCS.
    - Does your server run in a load limited environment (e.g. operating system quotas, virtual machine limits, etc.)? If so, these limits may need to be increased.
    - Are the file permissions in your cache directory set properly? If not, OJS/OCS will be unable to cache content, and this will severely decrease performance.
    - OJS and OCS support a number of additional caching mechanisms, e.g. the `web_cache` settings in `config.inc.php`. These may be useful for high load situations.

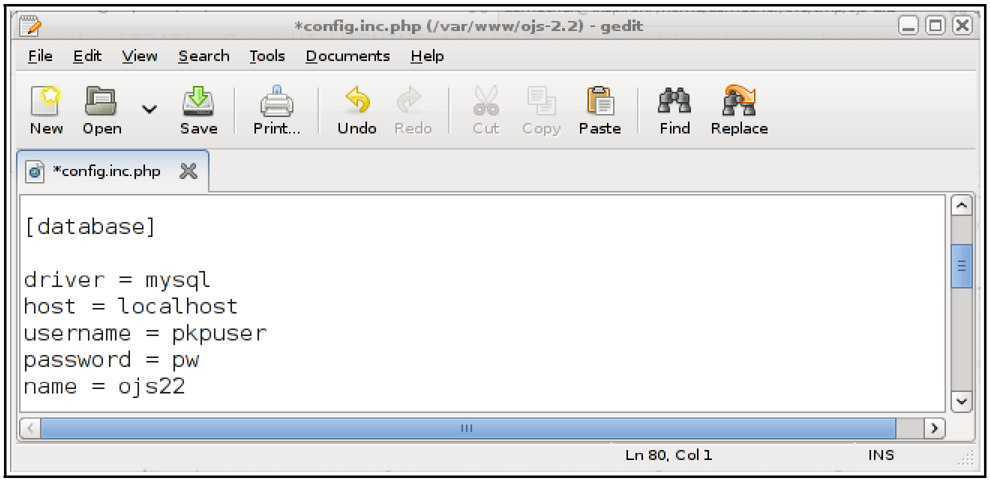
- If MySQL is experiencing heavy loads, you may need to investigate several aspects of the database:
    - If the file permissions in OJS/OCS's cache directory are not set properly, heavier load on the database may result. See above.
    - Check your MySQL logs to see if there are messages relating to damaged indexes etc.
    - If a particular index is missing from the database, performance will suffer. Find a page that exhibits particularly bad performance and temporarily enable the `debug` option in `config.inc.php`; this will cause all SQL statements to be displayed to the browser (note that this will affect all of the journal website's users while the option is enabled). Execute each of these statements separately and compare execution times to see which is causing the delay.

Import/Export

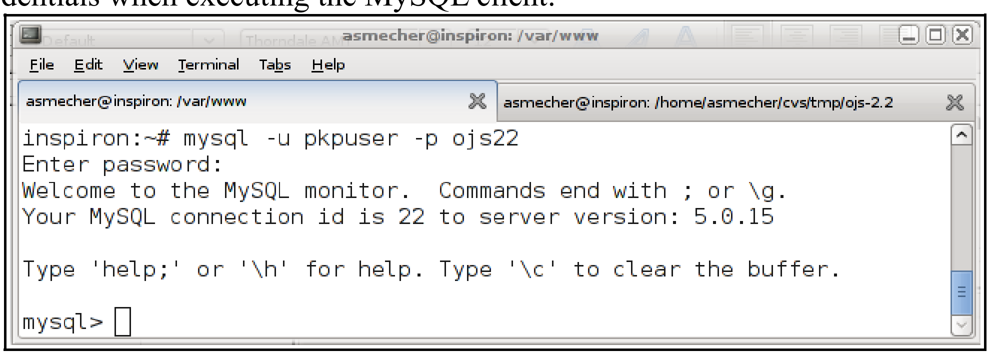
- Use the `xmllint` tool to validate your XML prior to importing.
- External files must be specified with a full URL (e.g. `http://...`) when using the native import tool via the web browser. Local filenames (e.g. `path/to/file.pdf`) are allowed when using the `tools/importexport.php` script.
- For OJS, the native DTD (`plugins/importexport/native/native.dtd`) specifies many additional details, such as the available root nodes for import documents.

Database Access

To access the database, find the access credentials in the `config.inc.php` configuration file:



Use these credentials when executing the MySQL client:



Once logged into the MySQL client, you may use the standard SQL commands, such as `INSERT`, `SELECT`, `UPDATE`, `DELETE`, etc. See the MySQL reference at http://dev.mysql.com/doc/refman/5.0/en/ for SQL syntax information, and the OJS Technical Reference from the Documentation link from the OJS web site (http://pkp.sfu.ca/ojs).

Common Database Tasks

It is possible to perform most tasks that will be needed for a successful conference or journal without needing to access the database directly. However, it may occasionally be necessary to fetch information from the database. See the “Database Access” section above for information on executing SQL statements; alternately, use a graphical interface such as PhpMyAdmin. (SQL statements can be executed in PhpMyAdmin by clicking the “SQL” icon.)

Determining a User ID

A user ID can be fetched from the database using one of the following SQL statements:

- If you have an email address for the user (e.g. pkp-support@sfu.ca):
  `SELECT user_id FROM users WHERE email = 'pkp-support@sfu.ca';`
- If you have a username (e.g. admin):
  `SELECT user_id FROM users WHERE username = 'admin';`

Password Reset

To reset as user's password, follow these steps:

- Determine the user ID (see above).
- Determine the password encryption method. See the `encryption` setting in the `[security]` section of the `config.inc.php` configuration file. This will be `sha1` or `md5`.
- Execute the following SQL to reset the password. For example, for user ID 15 with `md5` encryption, to reset the password to “newPasswordHere”:
  ```
  UPDATE users SET password=MD5(CONCAT(username, 'newPasswordHere'))
  WHERE user_id = 15;
  ```

Administrator Role Grant

All roles except Site Administrator can be assigned using the Journal Manager or Conference Manager's user interface. To assign a site administrator role to a user, follow these steps:

- Determine the user ID (see above).
- Execute the following SQL statement to assign a site adminstrator role (e.g. for user ID 15).
  - For OJS:
    `INSERT INTO roles (user_id, journal_id, role_id) VALUES (15, 0, 1);`
  - For OCS:
    ```
    INSERT INTO roles (user_id, conference_id, sched_conf_id, role_id)
    VALUES (15, 0, 0, 1);
    ```