EPrints Training: Plugin Scripting Exercises

Getting Started

Pre-installed EPrints repositories are provided for this session. Use the following parameters to access your training repository:

Repository hostname	
Repository URL	
Repository ID	
SSH login name	
SSH login password	

Importing test data

These exercises require you to have some records in your repository. If you had to erase your repository in an earlier session, you can import some test data by running the following command:

```
eprints> /opt/eprints3/testdata/bin/import_test_data REPOSITORYID
```

Plugin phrases

For some plugins, the text that EPrints displays in the Web interface needs to be defined in a phrases file. A sample phrases file for these exercises can be downloaded from:

```
http://www.eprints.org/services/training/resources/scripts/eprints3/plugins/plugin-exercises.xml
```

Before starting, make sure that this phrases file has been installed:

```
/opt/eprints3/lib/lang/en/phrases/plugin-exercises.xml
```

Export Plugin 1: Eprint titles

An export plugin that exports eprint titles

Change to the export plugins directory - this is where all EPrints export plugins live:

```
eprints> cd /opt/eprints3/perl_lib/EPrints/Plugins/Export
```

Create a directory to store your export plugins:

```
eprints> mkdir REPOSITORYID
eprints> cd REPOSITORYID
```

Using a text editor, enter the following into a file called `Titles.pm` (replace `REPOSITORYID` with the ID you have been assigned):

```
package EPrints::Plugin::Export::REPOSITORYID::Titles;

use EPrints::Plugin::Export;
@ISA = ( "EPrints::Plugin::Export" );

use strict;

sub new
{
        my( $class, %opts ) = @_;
        my $self = $class->SUPER::new( %opts );
        # Register plugin
        $self->{name} = "Titles (REPOSITORYID)";
        $self->{accept} = [ "dataobj/eprint", "list/eprint" ];
        $self->{visible} = "all";
        $self->{suffix} = ".txt";
        $self->{mimetype} = "text/plain; charset=utf-8";
        return $self;
}

sub output_dataobj
{
        my( $plugin, $dataobj ) = @_;
        # Convert - print title
        my $title = $dataobj->get_value( "title" );
        return "TITLE: $title\n\n";
}
1;
```

Test plugin (command line)

Run:

```
eprints> /opt/eprints/bin/export REPOSITORYID archive
```

You should see a list of all available export plugins.

Run:

```
eprints> /opt/eprints3/bin/export REPOSITORYID archive REPOSITORYID::Titles
```

Test plugin (Web interface - requires server restart)

- Load the home page of the repository in your browser
- Perform a search using the quick search box (top right)
- Choose your plugin from the list on the search results page and click *Export*

Exercises

- Familiarise yourself with the `export` script - find out what export plugins are available for exporting other datasets (hint: `user`, `document`, `subject`) and try them. Also investigate its counterpart - the `import` script. What import plugins are available?
- Change the plugin code so that it also prints the *year* and *month* for each item (hint: look at the BibTeX export plugin)
- Change the plugin code so that it also prints the *authors* for each item *one per line* (hint: look at the EndNote export plugin)
- Change the plugin code so that it also prints the *full citation* for each item (hint: look at Text export plugin)

Notes

- Note that you needed to restart the Web server in order to use the export plugin via the search interface. To improve the performance of the repository's Web interface, all Perl modules (this includes all plugins) are loaded into memory when the Web server starts. Therefore, whenever a new plugin is added, or an existing plugin is modified, the Web server needs to be restarted so that the plugin is reloaded.
- You may have noticed that, unlike the repository configuration files, all plugins are stored in the same directory (`/opt/eprints3/perl_lib/EPrints/Plugins`) and are available to all repositories running on the server. Plugins can be disabled for a specific repository by editing the plugin configuration file - investigate `/opt/eprints3/archives/REPOSITORYID/cfg/cfg.d/plugins.pl`

Export Plugin 2: Depositor activity

An export plugin that lists the depositor and deposit date for each item.

To save typing, the rest of the plugins for this session are available from:

```
http://www.eprints.org/services/training/resources/scripts/eprints3/plugins/
```

Download the plugin `Activity.pm` and examine it in a text editor (replace `REPOSITORYID` with the ID you have been assigned):

```
eprints> wget http://www.eprints.org/services/training/resources/scripts/eprints3/plugins/Activity.pm
```

Test plugin (command line)

Run:

```
eprints> /opt/eprints3/bin/export REPOSITORYID archive REPOSITORYID::Activity
```

Exercises

- Change the plugin code so that it also prints a column for the title of each item
- Change the plugin code so that the output is ordered by date deposited (hint: look at the RSS2 export plugin)

Notes

- Note that the plugin overrides the `output_list` method defined by the top-level export plugin `Export.pm`. Investigate `Export.pm` and find its `output_list` method - note how the main loop of this method passes each item in the export list to `output_dataobj`. Normally this is all that is required, but the Activity plugin overrides the `output_list` method in order to print the column headers *before* handing back to the default output_list method (`SUPER::output_list`).

Screen Plugin 1: Simple render

A simple render-only screen plugin

Change to the screen plugins directory - this is where all EPrints screen plugins live:

```
eprints> cd /opt/eprints3/perl_lib/EPrints/Plugins/Screen
```

Create a directory to store your screen plugins:

```
eprints> mkdir REPOSITORYID
eprints> cd REPOSITORYID
```

Download the plugin `Minimal.pm` and examine it in a text editor (replace `REPOSITORYID` with the ID you have been assigned).

Test the plugin (Web interface - requires server restart)

- Load the home page of the repository in your browser and log in
- Check that a link to your screen plugin is shown in the list of user tools
- Click the link

Exercises

- Change the plugin code to add a second level heading (h2) before the paragraph
- Change the plugin code to add a link to your homepage (or the EPrints homepage - `http://www.eprints.org/`) after the paragraph (hint: `render_link`)

Screen Plugin 2: Eprint overview

A (render-only) screen plugin that displays information about an eprint

Download the plugin `Overview.pm` and examine it in a text editor (replace `REPOSITORYID` with the ID you have been assigned).

Test the plugin (Web interface - requires server restart)

- Open the following URL in your browser:

```
http://REPOSITORYURL/cgi/users/home?screen=REPOSITORYID::Overview&eprintid=1
```

Exercises

- Experiment with different `eprintid` values in the URL
- Change the plugin code so that the citation of the eprint is also displayed
- Change the plugin code so that all documents attached to eprint are also displayed (hint: `get_all_documents`)

Notes

- EPrints has a number of screen plugins that display information about, and carry out actions on, eprints in the repository. These plugins are stored in `/opt/eprints3/perl_lib/EPrints/Plugins/Screen/EPrint`
- All Eprint-related screen plugins (including this Overview plugin) subclass the EPrint plugin (`/opt/eprints3/perl_lib/EPrints/Plugins/Screen/EPrint.pm`). Investigate this plugin and in particular the `properties_from` method - follow how this method reads the `eprintid` parameter from the URL, gets hold of the associated EPrint object, and makes it available as `$self->{processor}->{eprint}` (as used by the Overview plugin).

Screen Plugin 3: Toggle eprint field

A (action only) screen plugin that toggles the value of an eprint field

Download the plugin `Toggle.pm` and examine it in a text editor (replace `REPOSITORYID` with the ID you have been assigned).

Test the plugin (Web interface - requires server restart)

- Load the abstract page of an item in your repository e.g.
  ```
  http://REPOSITORYURL/1/
  ```
- Click the *item control page* link (bottom right)
- On the item control page, select the *Actions* tab
- Verify that selecting your *Toggle* action changes the value of the refereed field (check the value on the *Details* tab)

Exercises

- Change the plugin code so that the message displayed to the user is defined in a phrases file (e.g. `/opt/eprints3/archives/REPOSITORYID/lang/en/phrases/toggle.xml`) rather than hard coded into the plugin (hint: `html_phrase`)

Notes

- Defining the phrases that your plugin uses in a separate phrases file is good EPrints coding practice and allows:
  - ✔ the text displayed by the plugin to be easily changed without having to read the plugin code
  - ✔ your plugin to be used by repositories that support multiple languages (after translating the plugin phrases into the desired languages)
- Investigate the *Latest News* plugin as an example of a screen plugin distribution that comes bundled with its own phrases file (as well as a configuration file and stylesheet)
  ```
  http://files.eprints.org/141/
  ```

Screen Plugin 4: Update eprint metadata

A combined (render + action) plugin that displays the current value of a field and allows the value to be changed

Download the plugin `SimpleForm.pm` and examine it in a text editor (replace `REPOSITORYID` with the ID you have been assigned).

Test the plugin (Web interface - requires server restart):

- Open the following URL in your browser:

```
http://REPOSITORYURL/cgi/users/home?screen=REPOSITORYID::SimpleForm&eprintid=1
```

- Verify that selecting a value from the input form and clicking *Update* updates the value of the refereed field correctly (check the value on the *Details* tab of the item control page).

Exercises

- Change the plugin code so that it displays a warning message if you click *Update* without actually changing the value of the refereed field
- Change the plugin code so that it also allows you to change the publication status of the item (hint: use `render_name` to display the field names)