

# Welcome to the org.apache.forrest.plugin.input.projectInfo Plugin

## Table of contents

1 Apache Forrest - org.apache.forrest.plugin.input.projectInfo Plugin.....	2
2 Changes.....	2
3 To Do List.....	2
4 Release Notes.....	2
5 Developers List.....	2
5.1 Contributor List.....	3
6 Project Details.....	3
6.1 Multiple Projects.....	3
6.2 Project Indexes.....	4
7 Experimental Features.....	4
7.1 Configuration.....	4

## 1. Apache Forrest - `org.apache.forrest.plugin.input.projectInfo` Plugin

This plugin generates project info from various configuration files.

## 2. Changes

By maintaining a file called `status.xml` you can manage a list changes to your code base. This list can then be generated as a changes page and/or an RSS feed of those changes. For example, here are the [changes](#) for this plugin, here is the same list of changes as an [RSS feed](#).

It's possible to limit the displayed changes to a particular version number as well. For example, here are the changes for [version 0.1](#) of this plugin (and as an [RSS feed](#)).

If you want to only retrieve the changes for the most recent version of the project then you can do that too. Here are the changes in the [current development version](#) of this plugin (and once more as an [RSS feed](#)).

## 3. To Do List

The `status.xml` file can also be used to manage a list of todo items for the community. For example, here is a [psuedo todo list](#) for this plugin (our real todo list is managed in an Issue Tracker, one day we hope to add support for that here too).

## 4. Release Notes

To produce release notes you must maintain a `status.xml` file for your project and request a page with an URL such as `http://domain.com/releaseNotes_0.7-dev.html`, this will be produce the release notes for 0.7-dev.

If the version number ends with `-dev` a warning will be included in the generated page, informing the reader that it is a development version and therefore the list of changes is incomplete.

For a `status.xml` action to be included in the release notes it must have an attribute `importance="high"`. When writing actions in `status.xml` you should write them with the following two questions in mind:

- should it be `importance="high"`?
- will the action read correctly in the release notes?

The introductory text in the release notes comes from the (optional) element `notes` (a child of the `release` element).

### Note:

The same options with respect to retrieveing specific versions of the release notes are available as with the changes feature (discused above).

## 5. Developers List

The `status.xml` file can also contain a list of committers and contributors which can, optionally, be displayed as part of the changes file.

At the time of writing this functionality is quite minimal, being just a list of authors at the end of the changes document. However, in future releases it is intended that a more configurable output will be available.

## 5.1. Contributor List

It may be that some items have been contributed by authors other than those listed in the developer list. These are accredited in a `due-to` attribute of the `action` element in `status.xml`. A list of this contributors is made available at the end of the release notes for each version.

## 6. Project Details

Project details are generated from a project descriptor file. This file can be in any of the following formats:

- DOAP
- DOAP over ATOM

### Note:

We need your help in adding more formats.

The location of this descriptor file is specified in the `locationmap` with the hint `pproject.descriptor`. This defaults to the root of your projects content directory. To place your descriptor in a different location simply add a matcher like that shown below in your `projectInfo.locationmap-projects` file. This file is, by default, in the root of the `webapp` directory.

```
<match pattern="project.descriptor">
  <location src="[path/to/descriptor/file]" />
</match>
```

To retrieve the project details page request the file `/projectDetails.html`.

### 6.1. Multiple Projects

You can use the `projectInfo` plugin to pull together project information for multiple projects. For example, take a look at the project details for [Apache Forrest](#).

Using the `locationmap` you can retrieve this content from anywhere you want, for example, the DOAP file for the Forrest project comes from directly from SVN, thus we are always assured of it being up to date.

To retrieve the project details of another project make a request for `projectDetails/projectDetails.[PROJECT_NAME].html`. For this to work your `project locationmap` must have an entry like this:

```
<match pattern="project.descriptor.forrest">
  <location
src="http://svn.apache.org/repos/asf/forrest/trunk/site-author/content/doap.xml" />
</match>
```

Of course, if you have a central location for all your project descriptor files you could use a match like this:

```
<match pattern="project.descriptor.*">
```

```
<location src="path_to_descriptors/{1}.xml" />
</match>
```

**Warning:**

The indexing of projects (described below) does not currently work with wildcard matchers in the locationmap.

## 6.2. Project Indexes

The system will automatically create an index of projects from your projects locationmap file. For example, the "Projects" section of the navigation on the left is automatically generated.

**Note:**

There is considerable repetition in the indexes in this demo. The intention is to show the range of indexes that can be created. You should select the most suitable for your individual project.

**Warning:**

The indexes to be included in your site navigation only work correctly when generating a static site. You should only use page based indexes for dynamic sites.

## 7. Experimental Features

These features are operational, however, they are not fully developed and may change considerably before they become part of the official feature set. You can use them, but be prepared for changes, possibly without warning. If you do use these features we recommend that you join the developers mailing list.

### 7.1. Configuration

This plugin uses an experimental properties system that allows plugins to expose configuration information to the project. It is likely that at least some of these configuration options will eventually move into Dispatcher based contracts. In the meantime, you can use this config system to control some aspects of the display information. See the `forrest.properties.xml` file for a description of the config options available. To alter the configuration simply override these properties in your projects `forrest.properties.xml` file.