

# Project roles

## Table of contents

1 Overview.....	2
2 The roles.....	2
2.1 PMC Member.....	2
2.2 PMC Chair.....	2
2.3 Release Manager.....	2
2.4 ForrestFriday Coordinator.....	2
2.5 Issue Tracker Coordinator.....	3
2.6 Documentation Coordinator.....	3
2.7 Subversion Monitor.....	4
2.8 Legal Monitor.....	4
2.9 Developer.....	4

## 1. Overview

There are many tasks that need to be done to keep the project flowing. With the tasks being well-defined, the hope is that various people will be able to assist to carry out the role and so not rely on a couple of individuals to do all the work.

There are no roles for "leadership" of technical areas. We don't have the concept of leadership at the ASF. Rather these are "functional roles".

This is not intended to put pressure on people to force them to do work nor "pull their weight". If they do not manage to do a task, then someone else will be able to do it. Of course, shared load means a healthy project.

Most roles have minimal tasks and are not too onerous. For example, for the "Documentation Coordinator", the only job might be to publish the documentation every Friday. Sure it can be done at other times and other people can do it too, but at least it gets done once per week.

Having a person doing a role does not mean that everyone can leave it to them and rely on them. Anyone else can dive in and do the job (for example, publish the docs twice per week). The more people who are familiar with the role, the better. Perhaps the person who is primarily doing that role will notice that there can be improvements to how they are doing it.

If a person is dissatisfied with the way a particular role is being done then find a way to offer constructive criticism, perhaps by assisting or by enhancing the role documentation.

Most of the roles (except of course the Developer role) can only be carried out by PMC members because commit access is required.

## 2. The roles

These are the main roles, including the obvious ones. There are probably other roles that need to be defined also. Each role should have some continually enhanced documentation about it, so that any person can do the tasks.

### 2.1. PMC Member

This is well-defined in our [project guidelines](#) and in the top-level ASF docs.

### 2.2. PMC Chair

This is well-defined in our [project guidelines](#) and in the top-level ASF docs.

### 2.3. Release Manager

Tasks are defined in [How to Release Forrest](#)

Only one person can do this role, although other people can assist. There is a lot of knowledge invested in this role. Document it as much as possible.

### 2.4. ForrestFriday Coordinator

---

The tasks are defined in [ForrestFriday](#).

Each month someone needs to co-ordinate the IRC session and be the main channel operator, maintain the logfile and commit it to the events SVN.

Ensure that the session is summarised.

This role can possibly be done by someone who is not a committer. However, a logfile and summary need to be committed to SVN.

## 2.5. Issue Tracker Coordinator

---

Add new Components, e.g. for each new plugin.

Other general Admin tasks.

Add new committers to the forrest-administrators group.

Add/enhance [Filters](#).

Review issues to ensure that they are properly categorised. This is best done as soon as a new issue comes in.

Review the list of unscheduled issues.

Each month get the project to review the outstanding major issues that are scheduled for the upcoming release.

## 2.6. Documentation Coordinator

---

Publish the documentation at least once per week.

Various people make edits to the source files, but often the documentation is not generated and published. Also people add new entries to site-author/status.xml but the changes.html needs to be regularly generated from it.

This role is not actually about doing the documentation. That should be up to everyone.

Generating and publishing the main docs is very easy using a local forrestbot:

```
cd site-author
forrest -f publish.xml build
forrest -f publish.xml deploy
```

This builds the documentation locally then deploys it by committing it to the [forrest/site SVN](#). Then a cronjob on the server will automatically publish it. However, if instant turnaround is required, then do this:

```
ssh people.apache.org
cd /www/forrest.apache.org
svn update
```

See some general notes about managing [project websites](#).

Note that forrestbot does not remove docs from the forrest/site SVN (FOR-392). So need to manually 'svn delete oldDoc'.

Publishing documentation for a particular plugin is done by:

```
cd plugins/myPluginName
```

```
$FORREST_HOME/tools/ant/bin/ant deploy-docs
```

See further information in the [buildPlugin](#) doc.

## 2.7. Subversion Monitor

---

Ensure that svn:eol-style settings are "native" for all text files.

Ensure no line-endings issues.

Regularly run xml-tidy. (Not yet ready.)

Regularly run java-tidy. (Not yet ready.)

There are already some Perl scripts and other tools in the "[committers](#)" SVN in the "tools" directory.

## 2.8. Legal Monitor

---

Regularly run a script which verifies and inserts missing license headers to source files.

Regularly ensure that there are appropriate matching licenses to accompany each supporting product that we redistribute.

This should have continual oversight by the PMC as a whole, but the monitoring is something that needs to be done regularly, and definitely prior to each release.

There are already some Perl scripts and other tools in the "[committers](#)" SVN in the "tools" and "relicense/src/perl" directories.

## 2.9. Developer

---

The above roles are only for PMC Members. How can the Developers be involved? That is easy: do the Developer Role.

Help out with commenting on the Issue Tracker, fixing the issues, contributing to discussion, contributing patches, turning discussion into clear documentation.

This is incredibly valuable and will enable the project to grow. After time, you will probably be elected as Committer/PMC Member and when comfortable, can take on one of the Project Management Roles.