Project tasks

Table of contents

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 tasks 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 tasks".

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 tasks are not too onerous. For example, for the "Documentation Coordination", 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 task 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 task, the better. Perhaps the person who is primarily doing that task will notice that there can be improvements to how they are doing it.

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

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

2 The tasks

These are the main tasks, including the obvious ones. There are probably other tasks that need to be defined also. Each task 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 task, although other people can assist. There is a lot of knowledge invested in this RM role. Document it as much as possible.

2.4 ForrestFriday Coordination

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 task 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 Coordination

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 Coordination

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 task 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: See How to publish Forrest documentation.

2.7 Subversion Monitoring

The issue of whitespace is important in projects which have developers on both Windows and UNIX-based operating systems. The unnecessary differences obscure the real changes and so make it difficult for people to be aware.

Ensure that svn:eol-style settings are "native" for all text files. Encourage people to properly configure their SVN client. Detect such issues using a script like committers/tools/report_svn_text.pl See also some tips in FOR-124.

Ensure no line-endings issues. This is coupled with the abovementioned "svn:eol-style" issue. Detect such issues using a script like committers/tools/find_carriage_returns.sh

Regularly tidy the whitespace in xml files. See the script $FORREST_HOME/etc/tidy-xml.pl and the $FORREST_HOME/etc/tidy-config.txt configuration file. This is best done as part of the release process, when everyone should have already committed any outstanding changes. With such an operation, there is a high chance of conflicts. See also some tips in FOR-644.

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

2.8 Legal Monitoring

Regularly run a script which verifies and inserts missing license headers to source files. Review the output and fix issues. For example: committers/relicense/src/perl/ and other tools at committers/tools/ and Apache RAT (currently in Incubator) and some other suggestions at Apache Labs. See

also some tips in FOR-855 and FOR-123. Also see $FORREST_HOME/etc/relicense.txt and $FORREST_HOME/etc/relicense-avoid.txt files.

RAT is automatically run on our sources. See the reports via Gump and Buildbot. We are not being notified about errors, so need to look.

Regularly ensure that there are appropriate matching licenses to accompany each supporting product that we redistribute. (Use the filename pattern: product-version[.-]license.txt or uppercase.)

For each such license, also ensure that it is referred to in $FORREST_HOME/LICENSE.txt file. If their license required attribution, then add to $FORREST_HOME/NOTICE.txt file. Also replicate any NOTICE which accompanies a supporting product. See FOR-857.

See some guidelines at ASF Legal.

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.

2.9 Developer

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

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 tasks.