Samvera Community Principles

Overview

- Samvera can ultimately be successful and sustainable in the long run only if it is an open project; that is, it takes contributions from a community of developers across many institutions to enhance and support it.
- We are dedicated to maintaining a healthy climate for all community members, part of which is embodied in the Anti-Harassment Policy and the list of Samvera Community Helpers.
- We will work to balance progress on Samvera codebases with open community discussion and transparent decision making as coequal goals.
- All code for Samvera components is available through the Apache 2.0 open source license.

Technical Leadership

- Technical leadership of the project will be through a small group of proven developers who have demonstrated commitment to Samvera's progress and success (and have commit rights to the source code).
- Committers must be:
  1. technically adept
  2. constructive, positive members of the Samvera software community
  3. committed to producing useful, practical code for the community
- To become a committer, candidates must be…
  1. nominated by a current committer
  2. voted on and approved by a majority of the current committers
  3. committers may be voted out at any time by a (super?) majority of the other committers
- Before any code can be accepted, committers (and/or their institutions) must have signed a Samvera Community Intellectual Property Licensing and Ownership and lodged it with Samvera.
- Committers will have a regular meeting, usually in the form of a conference call, to coordinate development & direction.
- Releases will be vetted and controlled by a designated lead or leads. These roles may shift from release to release.

Code Contributions & Principles

Please refer to the Developers section of the wiki for suggestions about how to apply these principles in your work.

- The users of interest in, resources, and talent pool of the Samvera community will spread far beyond the developers on the committers list, and their institutions.
- Samvera encourages and will facilitate taking code from contributors from many sources.
- The structure of the source code management (currently using git) will facilitate incorporating and using code from many sources.
- Samvera committers will actively take code contributions from non-committers and incorporate it into the code trunk.
- Working code wins.
- You get what you give [getwhatyougive]
- All contributed code must have full test coverage before it is committed. The current testing infrastructure is RSpec for everything but Rails views, and Cucumber for for Rails views.
- Tests must be committed at the same time code is.
- All javascript features (where feasible) should be added using “progressive enhancement” or “unobtrusive javascript” styles. Meaning the basic feature should work without Javascript, and the JS should be added as a progressive enhancement, and added via externally linked js files, not in-line in html source as script tags, onclick attributes, etc.
- All bugs and development tasks will be tracked in JIRA.
- All code must be documented before it’s committed.

You get what you give

1. Samvera institutions commit to porting good functionality back to core. When implementing new functionality, try to do it in ways that lend themselves to re-use in other Samvera applications. When you set out to create new functionality, provision resources for contributing back.
2. What to expect from the community... Where to find extra help...

New Samvera applications should be posted as a publicly accessible showcase (either screencasts walking through novel features or publicly accessible demo with sample content and sample accounts with edit/admin permission).