Open Geoportal Developer Working Group (OGP-DWG) Governance

Version 0.1 - Last updated December 6, 2013

Mission Statement

The Open Geoportal Developer Working Group will design, implement, and write quality code for Open Geoportal.

The mission will be achieved by:

- Providing recommendations for features, architecture improvements, and code quality.
- Ensuring new releases of the software are completed on a timely basis.
- Supporting the initiatives of OGP Steering Group
- Performing outreach to the broader technical community
- Coordinating the development effort of institutions and individuals contributing to OGP

Overall project leadership is provided by the Open Geoportal Steering Group (OGP-SG) in collaboration with the metadata working group and the OGP-DWG. These three organizations are responsible for creating and supporting the OGP Development Roadmap. OGP-SG is responsible for prioritizing development tasks and providing scheduling guidelines. An organization that has representation on the OGP-SG will typically have an employee who is a committer, allowing organizations to both provide vision to the overall project and implement the features they need.

1.0 ROLES AND RESPONSIBILITIES

The OGP-DWG recognizes five roles: Users, Contributors, Committers, Release Manager(s), and Technical Lead(s). An individual may have multiple roles.

1.1 User

Users download, install, and configure an OGP instance to provide access to consumers of spatial data. Without users, there is no reason for the project. Users report bugs and make feature requests and suggestions. Users may receive invitations to take part in beta testing and surveys.

How to become one: Download the software from GitHub.

1.2 Contributor

Contributors provide source code, but do not have write access to the GitHub project source tree. In addition to code, contributions can be in the form of bug reports, documentation, and website content such as articles or FAQs. Integration of a Contributor's submission is at the discretion of the Committers and is an iterative, communicative process. For code to be integrated, a completed Contributor's License Agreement (CLA) is required (see below). Code

contributions should be accompanied by unit tests and documentation.

How to become one: Contribute in any of the ways described above. Contributors are strongly encouraged to package code submissions as a GitHub pull request from a forked repository.

1.3 Committer

Committers have commit access to the source tree, either for the individual modules they are working on, or global write permissions to entire GitHub-based source tree. They are responsible for making sure that code contributions do not break the build and enforcing the policies of the OGP-SG. Committers ensure that the quality of the software and the associated documentation remains high and the conceptual integrity of the project is maintained.

Each committer must complete and send in a CLA form (see below) to commit code. An organization that has a member with committer status may also have representation on the OGC-SG.

How to become one: A person who makes multiple contributions to the project may be nominated to become a committer. The steps include:

1. If you are interested in becoming a committer, please submit several useful pull requests.
2. Then, ask a current committer to nominate you to the committer team.
3. You will be asked to provide a statement of your experience and skills.
4. Current committers will discuss and vote on all nominations within a reasonable time period. See below for voting procedures.

1.4 Release Manager

The Release Manager's primary responsibility is to lead collaboration among Open GeoPortal developers to ensure timely releases. Their duties include fostering discussions to reach decisions, providing schedule guidelines, and managing cutting a release.

How to become one: A Release Manager will be selected by the Committers. All committers are expected to take a turn as the release manager.

1.5 Technical Lead

This person will ensure development work on OGP does not become stagnant and provide technical expertise for major OGP features. Part cheerleader, part task manager, the Technical Lead should "own" the various major systems within OGP so that they can provide documentation and answer questions from contributors and committers. This person is also the primary representative to the OGP-SG. Examples of responsibilities include:

- Respond to GitHub pull requests
- Non-release specific issues
- Maintain list of open issues, desired features, etc.
- Address unaddressed issues

How to become one: The Technical Lead will be selected by the Committers.

2.0 CONTRIBUTING CODE

Everyone is encouraged to contribute code to Open Geoportal. If you are not a committer, you can make a contribution by submitting a pull request from a contributor's fork via GitHub. The Technical Lead or another committer will contact you. If the pull request relates to a particular bug or requested enhancement then please provide this information. Code contributions should be accompanied by unit tests and documentation.

Contributor's submission is at the discretion of the Committers and is an iterative, communicative process. For code to be integrated, a completed Contributor's License Agreement (CLA) is required (see below).

2.1 Contributors' agreement

In order to accept a pull request we require either an individual or *corporate* Contributor's Licence Agreement acknowledging certain terms and conditions for its use. The purpose of a CLA is to ensure that the guardian of a project's outputs has the necessary ownership or grants of rights over all contributions to allow them to distribute under the chosen licence (Wikipedia 2013). The CLA also ensures the contributor that their work will remain open under the XXXX license agreement. Currently, the guardian of Open Geoportal is Tufts University.

Once this agreement has been completed and a pull request is accepted then the differences will be applied to the original source code or documentation by the project team and committed to the source code repository on your behalf.

3.0 DECISION MAKING PROCESS

Committers will strive to achieve consensus. A vote will be passed if a majority of committers vote +1. We will use lazy "yes" voting, that is if a committer does not vote in the defined voting period, their vote will be affirmative. A minimum of XXX +1 votes are needed for a motion to pass. A voting period will typically be one week.

Often, developers will work on tasks selected by their home institution. Hopefully, this results in a pull request so the feature can be shared with the broader community. They should discuss design and architectural issues with the broader OGP technical community. The follow are some examples of decisions that should be made collaboratively within the OGP-DWG:

- Adopting or eliminating new libraries or dependencies
- Schema changes
- User interface changes
- Refactoring code
- Management/Tracking features
- Tools Sets
- Infrastructure (Libraries, plug-ins, etc.)
- Release Manager
- Technical Lead
- Periodic Open Meetings

Requested new features that are not in the Roadmap should be voted on by the OGP-DWG. If it is a type of feature that requires OGP-SG approval (UI changes, architectural issues, etc.), the OGP-DWG will make a recommendation to the OGP-SG for approval subsequent to the vote. Those that do not require a vote are adopted.

4.0 FEATURE REQUEST AND BUG TRACKING

Feature requests and bug tracking are encouraged and can originate from anywhere in the OGP community. Most feature requests and issue tracking is managed by the OGP-SG. Tasks are placed on the roadmap by collaboration between the OGP-SG and the OGP-DWG. Changes that affect the user interface may require a more detailed review. Features added as external modules naturally require less review and collaboration. Changes related to improving the quality of the code, issue tracking, bug fixes, simplifying installation, etc. are managed internally by the OGP-DWG. Features no approved by either the OGP-DWG aor the OGP-SC can still be contributed as community modules if properly packaged.

4.1 Mailing lists

The OGP-DWG maintains a public mailing list for technical questions and discussions. The developers mailing list is at https://groups.google.com/forum/#!forum/opengeoportal. Note that there are additional mailing lists for the OGP-SG and other OGP affinity groups.

Architecture [Dave]

-New features should be categorized in some way that influences/determines the amount of involvement required by the Steering Group. For example: a feature that requires either a UI change, a substantive change in end-user experience (like changing the Solr query), or a substantive increase in resources should require Steering Group approval. Also changes that would substantively affect the install base (like changes to the Solr schema). Changes to code structure, libraries used, etc. generally should not.

Background Information: Governance Models And Resources

Karl Fogel's Producing Open Source Software at http://producingoss.com/

Governance models: http://www.oss-watch.ac.uk/resources/governanceModels

Much of the above come from Taverna, an academic open source project: http://www.taverna.org.uk/about/legal-stuff/taverna-governance-model/

Background on CLA: http://en.wikipedia.org/wiki/Contributor_License_Agreement

Apache's CLA: http://www.apache.org/licenses/cla-corporate.txt