

Best practices for the OpenBMC community

Kurt Taylor (krtaylor)
August 14, 2018

Part 3: Contribution



OpenBMC

Contribution



OpenBMC

- EVERYONE can contribute
 - Code, Reviews, Testing, Bug Triage, Documentation, Email discussions, Answering IRC questions, Organization, Scheduling...
- Find the existing work area, connect with others, join an existing team
 - Reach out on IRC or in email
- Form a workgroup to accomplish something new
 - Email description of the workgroup, what you want to accomplish
 - Form open teams around a feature, not around a brand or manager
 - If you are the only member, is it really necessary?



Contribution

- Code
 - Make SURE you have signed the CLA
 - Maintainers will not merge code from anyone that has not signed a CLA
 - Learn Git
 - <https://git-scm.com/doc>
 - Read contribution guidelines
 - <https://github.com/openbmc/docs/blob/master/contributing.md>
 - Tell everyone (introduce yourself)
 - That you are learning about an area, any pointers
 - Learn the code, do your homework
 - Ask for **clarification**, not instruction
 - Follow up with documentation refinements for others that will follow
 - Learn Gerrit
 - <https://review.openstack.org/Documentation/intro-quick.html>
- Patch Sets
 - Limit the addition of reviewers
 - The reviews should happen normally, if not then ask someone in IRC if you can get a review – limit this behavior, called “begging a review”
 - It is acceptable to add a peer developer in the team that has been working with you on the feature

Code Reviews



OpenBMC

- Start in an area that you are comfortable with
- Don't be a yes-person, just to boost your review count (metrics)
 - Commenting "LGTM" is fine, but if every one is always LGTM, then your reviews aren't really doing anyone any good
 - Shoot for 50-70% positive for a quality review reputation
 - BUT...
 - Don't neg a comma in a comment, or a "teh" this should be a comment
- If you review a patchset, follow up as soon as you can
 - Be available to clarify or discuss in IRC
 - Reply quickly in the patchset comments with answers
- You WILL get grief
 - Be prepared to reply with facts and not emotions
 - Do not have a discussion in comments, take it to IRC or a weekly meeting to get a broader audience

Contribution



OpenBMC

- Issues/Bugs
 - Testing is good for everyone!
 - If you find a problem, document it *somewhere*
 - Ask in IRC for guidance
 - Feature/WaD/Bug
 - Designed that way
 - Clarify what and why in the documentation, or comments
 - Bug – Open an issue
 - Be sure and include enough information for another developer to be able to reproduce
 - Include any messages, log snippets, dumps
- If you write a bug, **you** own it's management
 - Be responsive to questions for more info
 - Assist anyone that is trying to reproduce or fix
 - Seek others that may have seen something similar
 - Make sure it gets assigned to the proper component or project
 - Document **timely** resolution

Contribution



OpenBMC

- Documentation/Wiki
 - New experience invaluable and unbiased
 - Requirement for new features
 - Continual update, no stagnant docs
 - Quick and dirty documentation is better than none, iterate
- Infrastructure
 - Help manage tools needed to run the community
 - Wiki, repos, bots, IRC, services, logs, email
 - New tools, investigation and proposal
 - Simplification of existing services
 - Automation of processes
- Mentoring
 - Help someone in IRC or email
 - Improve developer experience
 - Reduce bottlenecks and bureaucracy
 - Ease onboarding new contributors