

# StarlingX PTG

Denver Sep 2018

# Agenda

9:00 - 10:00 AM	Planning process - feature life cycle, defect process... (Bruce)
10:00 - 10:30	Governance (Bruce)
10:30 - 11:30	Release planning - Oct18 and Mar19 (Ghada)
11:30 - 12:00	Break / lunch
12:00PM - 1:00	Rebasing OpenStack (Brent)
1:00- 1:30	Python2->3 (Bruce/Dean)
1:30 - 2:30	Multi-OS - content, feature impact, builds, packaging, etc.. (Brent/Saul)
2:30 - 3:30	Infrastructure Containerization (Brent)
3:30 - 4:00	Build - Mirror construction, src rpm build, (Saul)
4:00 - 4:30	Keystone Edge Architecture (Greg)
4:30 - 5:00	Project decoupling update (Brent)
5:00 - 5:30	Cross project collaboration
5:30 - 6:30	Happy Hour / Break / dinner
7:00 - 10:00PM	Storyboard grooming

# Planning process

# Feature Planning Process

Proposed goals of the planning process

- We will plan our work following the Four Opens
- The process should
  - Be open and transparent but not create spam
  - Be collaborative but not require additional meetings
  - Be formal but not painful
  - Be as light as possible
  - Be as easy as possible for new Feature ideas to be submitted to the project team
  - Be aligned with our Governance and the roles it defines
  - Support a feature development life cycle
  - Be designed to leave a permanent record of the architecture and design discussions

# Terminology

**Bug fix** - a change to the software to correct a defect.

**Enhancement** - a change to the software for internal reasons (e.g. refactoring to improve testability, python2->3, etc...)

**Feature** - a change to the software that is visible to end users

Proposal:

1. Handle Bug fixes with LaunchPad. No other process is necessary.
2. Define one process for both Enhancements and Features. In this presentation I will use the word Feature to mean both.

# Models to consider

## 1. Our current practice

- a. Ad hoc and inconsistent.

## 2. Linux Kernel

- a. Email based. No formal specs. Hierarchy of maintainers who approve changes for their subsystem

## 3. Current OpenStack practice

- a. <https://docs.openstack.org/contributors/common/releases.html#release-schedule-and-planning>
- b. Connected to the OpenStack release cycle - milestones and code freezes
- c. Some new features are reviewed at PTG meetings, some have specs, not all projects behave the same way. Projects with higher load/smaller bandwidth tend to drive more closely to milestones
- d. The process and workflows are not consistent across all projects except for the final code freeze date and drive to release
- e. Challenges in coordinating features that span multiple projects

# Feature Lifecycle

- Features follow a common lifecycle. Some but not all projects explicitly track features through each step
- Common steps:
  - New - this is a new idea from a user, a developer or other project stakeholder
  - In review - New features need to be reviewed, scoped and estimated by the project team.
  - Approved- The feature has been reviewed and approved for implementation
  - In progress -The feature is being worked on
  - Complete - The feature has been fully implemented and tested
  - Released - The completed feature has been formally released
  - Rejected - The project team has decided to not implement the feature
  - Deferred - The project team has decided to implement the feature at some later point

# Issues to be addressed by the planning process

- How are Features tracked through the lifecycle?
- How is information about Features communicated to the team and project stakeholders?
- How and who (and if) reviews and approves Features?
- How are Features mapped to Releases?



# Scale

Not all Features have the same scale and may not require the same process

<b>Scale</b>	<b>LOC</b>	<b>Complexity</b>	<b>Projects impacted</b>
Small	< 200	Low	1
Medium	< 2,000	Medium	1-5
Large	< 20,000	High	Many

- Features with low LOC count but with High Complexity or broad Project impact might be Medium
- Features with high LOC count but with Low complexity and Project impact might be Small

# Feature planning proposal p1

- The project shall track all Features in a database
  - In the end, any discussion on this topic turns into a discussion on tooling.
  - Do we use Launchpad? Bugzilla? Jira? Trello?
  - Do we invest in improving Storyboard?
  - 29 Free/open tools described:  
<https://blog.capterra.com/free-open-source-project-management-software/>
  - Many of those tools are “freemium” and require \$\$\$ for large teams (like ours)
  - DECISION: Continue using StoryBoard and invest in improving it.
- Anyone on the project can enter a new Feature into the DB
- TSC members should take on the responsibility of entering new Features into the DB for project stakeholders who can not.
  - For instance for requests from users on the mailing list, or from internal management / sales teams, or whomever.
  - No idea should be rejected at this stage, all input and feedback is valuable

# Feature planning proposal p2

- For a New Feature to move to In Review, the TSC shall:
  - Review the Feature themselves and either:
    - Assign the Feature to a Project Lead (or Leads) for scoping
    - Move the Feature to Rejected (gently, provide feedback) or Deferred
- A Feature moves from In Review to Approved once it has been reviewed, scoped & analyzed and the Project Lead(s) agree to a target Release:
  - Each Feature shall have a spec, written by the Project Lead(s) or their delegates
  - The Spec should be posted to the email list for broader review
  - Specs for Small scale projects can be a paragraph in a tool.
  - Specs for Medium/Large scale Features could be either stored in the tool database or checked into the repos. We should pick one approach for all Specs.
    - Once again, we need to figure out tooling. Suggest that we check Specs into stx-specs/TBD DEAN/<feature name>.txt. Team to provide a basic template.
    - Spec review should be in gerrit and in email.
  - The Project & Team Lead(s) for the Feature jointly decide the Feature's target release and update the Release team (e.g. by tagging the DB entry).

# Feature planning proposal p3

- A Feature moves from Approved to In Progress once one or more people start working on it.
- Once all of the work to fully implement and test a Feature is complete, it moves to Complete.
- Features that have an implementation effort that spans multiple releases should be either done in a feature branch or implemented so as to not de-stabilize master.
- The Release team will work with the Project Leads during the release cycle to monitor release content and address deviations (push to next release, etc..)
- Features that make it into a Release are moved to Released in the tool.

Action: Consider defining a set of tags to use to show these state transitions in StoryBoard. AR Bruce & Ghada.

# Feature planning p4

- Do we have a planning cycle? Milestones?
  - Right now we have a Code Freeze date and a Release date with no intermediate milestones.

Open issue: Do we add additional milestones? We have been doing monthly builds but these are time based snapshots with no context attached to them.

Open issue: Setting priorities - coupled with resource allocation. Small scale features can be prioritized by the Project & Team lead. Larger scale features should be prioritized by the TSC. Priorities should reflect resource assignments - something can't be important but not assign to devs.... Impact to the existing code base should be considered

What does it mean for a Feature to be Done? Like Agile Acceptance Criteria, do we want to specify those in the spec itself? Hold PLs and TL's accountable?

AR Team - how can we arrange for open testing of StarlingX? Is this just 3rd party CI???



# Governance

# Governance

Our initial Governance document is on the StarlingX Wiki

[https://wiki.openstack.org/wiki/Starlingx/Initial\\_Governance](https://wiki.openstack.org/wiki/Starlingx/Initial_Governance)

The goal of this session is to review the document, identify any issues that need to be resolved, resolve them if possible quickly and work them offline if not



# Governance

The StarlingX project is governed according to the OpenStack Foundation's "four opens", which are open source, open design, open development and open community. Technical decisions are made by technical contributors, technical leaders and by a representative Technical Steering Committee. Our community is committed to diversity, openness, and encouraging new contributors and leaders to rise up.

# Governance

StarlingX is both a development project and an integration project. It includes new services that provide important features and combines them with components from many other Open Source projects into a complete Edge Cloud solution. To help manage the complexity of the project, we have divided the project up into several sub-projects, each with project and technical leadership, to help distribute the overall work and to acknowledge in the community that there are multiple ways to contribute to the project. The sub-project lifecycle is managed by the project's Technical Steering Committee who approve the creation of new sub-projects and the retirement of sub-projects that are no longer active.

# Governance

StarlingX is a brand new project and is in an initial "bootstrapping" phase in which the leadership positions will be appointed. All leadership positions will transition to be elected by the project's Contributors within one year.

# Governance (CHANGED)

## **Contributors**

A Contributor to StarlingX is someone who has made a Contribution to the project within the last 12 months. Contributions can include merged code, test or document submissions, or serving in a leadership role as defined below. All Contributions are welcome and will be accepted based on their technical merit.

The project's Technical Steering Committee can grant Contributor status for other contributions at its discretion. Contributors are eligible to vote in elections for Technical Steering Committee positions and for the leadership roles defined in this document.

# Governance

- Does this need to change to allow for more diversity amongst the Core Reviewers? No change needed.

## Core Reviewer

Core Reviewers are active Contributors and participants in a sub-project that have the additional responsibility to review the changes proposed to the sub-project, to ensure that approved changes are aligned with the project's design & architecture, and meet the project's quality requirements. Core Reviewers have the ability to approve code to be merged into the StarlingX repositories. Core Reviewers for a sub-project are appointed by the sub-project Technical Lead with input from other StarlingX Core Reviewers. Contributors can become Core Reviewers for multiple sub-projects.

# Governance

- When are the Technical Lead elections?
- Hold the TL elections annually (Sep 19) and stagger with the PL elections (April 19)
- Update this based on the Planning process e.g. approving specs

## Technical Lead

A Technical Lead in StarlingX is a Core Reviewer who has additional responsibility for guiding the overall technical direction of one or more of the sub-projects, under the overall technical guidance of the Technical Steering Committee. Technical Leads are responsible for resolving disagreements between the sub-project's Contributors and Core Reviewers. The initial Technical Leads are appointed to one year terms at launch by the Technical Steering Committee but will be fully elected by the sub-project's Contributors on an annual basis. Contributors can be Technical Leads for multiple sub-projects.

# Governance

- When are the Project Lead elections? (April '19)
- This part of our Governance is the biggest difference between us and usual OpenStack practice. And it's a good thing to separate the P and T from the PTL role.
- We seem to be turning this into a status reporting role - is that what we want? Yes, we want someone tracking and reporting status
- TL - inward facing. PL - outside facing, requirements gathering and reporting results, handling outside communication, project ambassador.
- Need to make the "break work down" more open and collaborative - the PL should facilitate, work with and help the team manage their work and work lists.
- AR Bruce update this entirely
- Needs to include R&Rs from planning process

## Project Lead

A Project Lead in StarlingX is responsible for the sub-project's work item list and facilitating the four opens within the sub-project. The Project Lead works with the Technical Lead to break down large work items for the team into Stories and Tasks. The Project Lead can help guide Contributors to the work items most needed by the sub-project, as defined by the Project Priorities established by the Technical Steering Committee. The initial Project Leads are appointed to one year terms at launch by the Technical Steering Committee but will be fully elected by the sub-project's Contributors on an annual basis. Contributors can be Project Leads for multiple sub-projects.

The same person can become a Technical Lead and Project Lead for a StarlingX sub-project.

# Governance

- Missing content here for managing sub-project life cycles - Bruce to add
- Change max to 7 for initial TSC. Allow it to grow following the governance change process
- Need additional clarity on the TSC members roles and responsibilities - add Planning items
- Add: The TSC sets project priorities in collaboration with the teams
- Need to define how the TSC voting process works - simple majority, super majority, consensus?
  - The TSC should seek consensus whenever possible
  - Voting should have a quorum requirement (5 members present)?
  - Most voting on normal technical issues should be majority
  - New project creation and governance changes should be 2/3s

## Technical Steering Committee part 1

The Technical Steering Committee is responsible for architectural decisions and making final decisions if sub-project Core Reviewers, Technical Leads or Project Leads disagree. It defines the overall project architecture and sets the overall Project Priorities. It will be comprised of 9 members who will be appointed at Launch but fully elected by the Contributors within the first year.

The initial Technical Steering Committee members will be:

Brent Rowsell and Ian Jolliffe from Wind River

Dean Troyer and Saul Wold from Intel

TBD...

We are actively recruiting for additional Technical Steering Committee members



# Governance - update with 7 seats

## Technical Steering Committee part 2

In April 2019, 4 of the 9 seats will be up for election by the project's Contributors. Anyone who is a Contributor to the project will be eligible to run, and anyone who is a Contributor is eligible to vote. In October 2019, the remaining 5 seats will be up for election. The elections will continue on this staggered cycle (4 seats and 5 seats) every six months in order to allow new leaders to rise up, but also ensure some consistency across the terms. There are no term limits, but in order to encourage diversity, no more than 2 of the 9 seats can be filled by any one organization. The Technical Steering Committee will meet regularly in an open forum with times and locations published in community channels.

AR: How do we decide which TSC seats stand for re-election? Let the initial TSC stand for one year. Then start the stagger with one rep from each initial company serving 12 months (if not re-elected) and one 18 months

# Governance

## **Technical Steering Committee part 3**

The Technical Steering Committee can elect a Chair at its discretion. Meetings will be hosted and facilitated by the OpenStack Foundation.

The exact size and model for the Technical Steering Committee will evolve over time based on the needs and growth of the project, but the governing body will always be committed to openness, diversity and the principle that technical decisions are made by technical contributors.

# Governance

- Add a link to the page that defines how OpenStack manages elections
- Ballots will be distributed to each Contributor's primary email address

## Elections

All elections for leadership positions in StarlingX shall follow standard OpenStack procedures and methods. In the event that a candidate runs unopposed for a position, the TSC can waive a formal vote. Membership in the Foundation itself is not a requirement for holding an elected position though it is preferred. Elections are appointing an individual to a position in the project, not a company or organization. Individuals are expected to continue to support the project in the event of career changes unless they notify the project that they are resigning their position.

# Governance

## **New Section - How to change our formal Governance**

OpenStack keeps their formal TSC charter in a repo. Only the chair of the TSC can approve changes to it.

Changes should be the result of an vote - in the TSC? By the Contributors?  
Simple majority? Super majority? Consensus?

Do we allow any Core Reviewer to +2 a change? Do we limit change approval to the TSC? The TSC Chair?

AR: The formal governance document should be controlled in a (new) repo. Votes should be in the TSC and require 2/3s approval but the TSC should strive for consensus.