

Tungsten Fabric PRC Launch

Nov. 2019



The Community

Governance

Technical Steering Committee :: Overall Project Governing Body

This primary interface to LF and LF-N for project-wide issues is divided into two permanent working groups:

- Meets every Thursday 10am PT

Community Working Group

Focused on business governance, the CWG handles policy, marketing, and budget

Technical Working Group

Focused on technical governance: use cases, architecture, lifecycle, and subprojects

Working groups within the TWG outlined on the next slide

Community Calendar :: <https://tungstenfabric.io/community/>

TWG Working Groups

Architecture Review Board (ARB)

Architecture consistency

- Define detailed Architecture
- Review Design Specs and Code
- delegated by Tech WG

TBD

Infrastructure Working Group (IWG)

Technical backlog

- Build a community CI
- Define documentation and test requirements
- Meets every other Tue at 8am PST

Marketing Working Group (MWG)

Interface to LF-N Marketing Adv. Council (MAC)

- Marketing plans and outreach
- Cross Community coordination
- Meets every Tue. 1pm PST

Documentation Working Group (DWG)

User and developer documentation

- Release Note - Lenovo
- Feature Docs – Juniper/Lenovo
- Developer Docs - General Community
- Meets every Wed. 11am PST

Landing Page → www.tungsten.io

Everything is here → <https://wiki.tungsten.io/>

- [Governance](#)
- [Contributor License Agreement](#)
- [Demos](#)
- [Docs Team](#)
- [Marketing](#)
- [Release](#)
- [Summits and Events](#)
- [Tech](#)

Governance

Created by Paul Carver, last modified by Edward Ting on Nov 03, 2019

TSC Voting Members

Technical Representatives

@ Sukhdev Kapur - Juniper sukhdev@juniper.net

@ Prabhjot Singh Sethi - ATS prabhjot.lists@gmail.com

@ Joseph Gasparakis - Intel joseph.gasparakis@intel.com

Open

@ Edward Ting - eting@cloudasoft.com

Community Representatives

@ Randy Bias - Juniper rbias@juniper.net

@ Ian Rae - CloudOps irae@cloudops.com

@ Jim St. Leger - Intel jim.st.leger@intel.com

@ Abhijeet Singh - AT&T AS753D@att.com

@ Darien Hirotsu - SDN Essentials /
Redapt darien@sdnessentials.com

Recurring Teleconference Meetings

[Technical Steering Committee](#)

[Infrastructure Working Group](#)

[Marketing Working Group](#)

[Documentation Project](#)

Community resources

- Slack — tungstenfabric.slack.com
 - #Dev – developer focus
 - #Documentation – software documentation
 - #General – general discussion
 - #Users – user focus
- Mailing Lists — <https://lists.tungsten.io>
 - 'announce' – everybody needs to subscribe
 - 'Conference' – event and conference proposals/discussions
 - 'arb' – architectural review board
 - 'cla' – submit CCLA/ICLA
 - 'dev' – developer focus
 - 'discuss' – general discussion
 - docs' – documentation related
 - 'marketing' – events and other outreach
 - 'security' – Security bug escalations
 - 'tsc' – Governance

Community resources

- Web — www.tungsten.io/community
- Blog — <https://tungsten.io/blog/>
- LinkedIn: <https://www.linkedin.com/groups/6517760>
 - Follow Randy Bias: <https://www.linkedin.com/in/randybias/>
- Twitter
 - Tungsten Fabric - [@tungstenfabric](https://twitter.com/tungstenfabric)
 - Randy Bias - [@randybias](https://twitter.com/randybias)

Technical resources

Github: <https://github.com/tungstenfabric>

- Repos are being migrated from <https://github.com/Juniper>
- Meanwhile, if you need any code: <https://github.com/Juniper/Contrail->*

Dockerhub: <https://hub.docker.com/u/tungstenfabric>

Gerrit

- Current – <https://review.tungsten.io/>

Bugs and Blueprints

- Current – <https://jira.tungsten.io/>

How to Contribute Code

The big picture



Sign up...

...Then submit
your blueprint



Sign up for mailing lists, Slack, Gerrit, and Launchpad, plus sign the CLA.

Then, submit your blueprint via Launchpad. Notify the #dev channel on Slack, as well as both the “dev” and “arb” mailing lists.

The big picture

Once your blueprint is approved and greenlighted for a particular release, you're ready to write your code and post for review to review.opencontrail.org.

JIRA: <https://jira.tungsten.io/secure/Dashboard.jspa>

Blueprint is a story, at least. Is there a template?

1. Submit your CLA

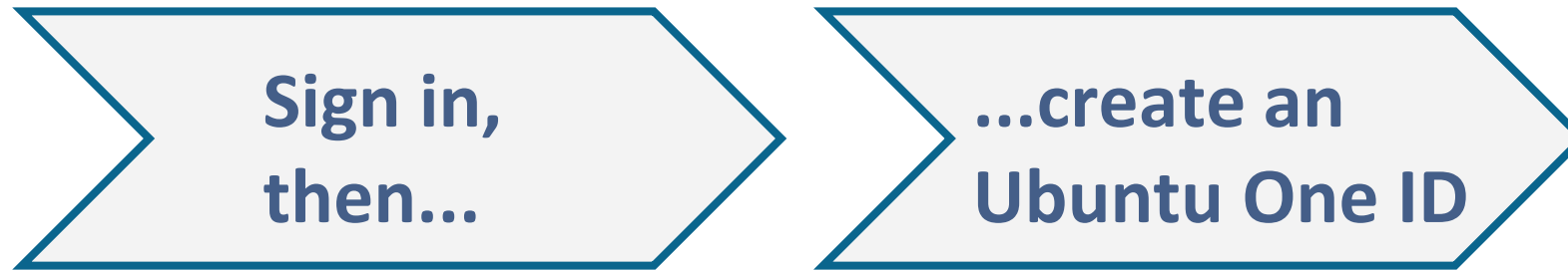
On the Google drive, there are two CLAs:

- Corporate Contributor Licence Agreement (CCLA)
<https://wiki.tungsten.io/download/attachments/1409065/Tungsten%20Fabric%20-%20CCLA%20Template%20v1.2.docx?api=v2>
- Individual Contributor License Agreement (ICLA)
<https://wiki.tungsten.io/download/attachments/1409065/Tungsten%20Fabric%20-%20ICLA%20Template%20v1.2.docx?api=v2>

Choose the right one for you, sign, and submit it.

2. Create a Gerrit account

Create a Gerrit account on review.opencontrail.org



Email cla@lists.tungsten.io with your CLA to get access to Gerrit.

Is it a feature, or is it a bug?

I'm working
on a BUG

I'm working
on a FEATURE

It all starts at Launchpad: <https://launchpad.net/opencontrail>

Attend a Tuesday meeting ask TC to accept the bug fix.

Submit a blueprint

<https://launchpad.net/opencontrail>

Blueprint is a use case. Here's a sample:
<https://blueprints.launchpad.net/opencontrail/+spec/ip6-arpa-zone>

Pay attention to the code completion deadlines! www.tungsten.io/community

What is a blueprint?

A blueprint is a use case. Here's an example:

<https://blueprints.launchpad.net/opencontrail/+spec/ip6-arpa-zone>

Ask yourself...

- Does the community need it? Does it fit our current architecture?
 - Alternative implementations not a good idea; if current implementation is broken, jump in and fix it.
 - Technology placement for its own sake is not welcome. All code contributes to the whole.
 - Simple, coherent, scalable, high performance, production
 - We are not a universal SDN big tent. Read the architecture, follow the architecture.
- Provide just enough detail for review, and save the implementation details for the spec.
- A majority vote means approval.

How Blueprints get approved

Blueprints are reviewed and approved by the TC

Tuesday meeting—get on the agenda

Mind the blueprint deadline for the current release!



After your blueprint is approved, create a spec

A specification (spec) is a detailed feature design. Here's an example:

[https://review.opencontrail.org/#/c/37214/3/specs/ipv6 reverse zone for vdns.md](https://review.opencontrail.org/#/c/37214/3/specs/ipv6_reverse_zone_for_vdns.md)

Specs can be approved by one member of the ARB. Start that process by sending an email to arb@lists.tungsten.io. Contact members on Slack.

Once your spec is approved, commit your code and test it (unit and integration) by the code completion deadline:

- Create a commit bug on Launchpad.
- Do your tests and fix your code.
- Submit a review request on Gerrit (review.opencontrail.org).
- Ask dev channel for code review on Slack. << VERY IMPORTANT!

Writing a great spec

Explain how the code fits the current architecture.

Explain how your code is simple, coherent, scalable, high performance, and production ready.

Specify all external and the most important internal interfaces, design, algorithms.

Show how will you test, and have a test plan ready.

Approval responsibilities

The TC is responsible for approving blueprints.

The ARB is responsible for approving specs.

The ARB, PTL and core contributors are responsible for code reviews.

Get help from Joseph Gasparakis, Suhkdev Kapur, and Paul Carver via Slack.

Release schedule

Releases goals and deadlines are set by TC, published on Slack and at www.tungsten.io/community.

Train model...

Releases go out on time, so features that don't make it catch the next release. You'll need a new ticket, and TC must approve your blueprint for next release.

CI and Test infrastructure

All community-wide tests run through CI.

For check-in/unit tests, Gerrit kicks Zuul, which does a build, then Zuul goes to OpenStack and runs the build.

Quick test are done using an all-in-one install (installed on a single VM), nested OpenStack-on-OpenStack. All tests run in a nested virtualized environment.

CI and Test infrastructure

For feature/integration tests, nightly tests run on the same nested OOO environment. Developers should commit feature tests.

Info on test scripts:

<https://github.com/Juniper/contrail-test/blob/master/README.md>

In addition to community tests, the commercial vendors will run their own tests: Juniper, Mirantis, Lenovo teams will report bugs to the community.

Code Acceptance and Maintenance: Fix Your Bugs!

Code with outstanding bugs will be reverted.

All code must be production ready

- no bugs
- stable
- scalable
- high performance (supports wire speed)

Only incubation/experimental subprojects are allowed to keep buggy code in the tree.

Docs and Help

Doc repo: <https://github.com/tungstenfabric/docs>

If there is something missing, or unclear, post a message to #dev channel on Slack best place for help.

Getting started: <https://github.com/tungstenfabric/docs/blob/master/Contributor/GettingStarted/getting-started-with-opencontrail-development.md>

Downstream Releases and Community Code

Currently, three downstream distros: Juniper, Mirantis, Lenovo

All use open core model

Community code which passed through QA cycle will be present, but may not be officially supported in a given commercial distribution.