



DLF

NETWORKING

Virtual Technical Meetings

Transparency of Tungsten Fabric Community

Szymon V. Gołębiewski

Marek Chwal

- Structure of TF Community
- Modules of TF Product and project structure
- Release process principles
- Documentation process
- How to become a TF member



Structure of TF Community

The Tungsten Fabric Project is an open source project to build all the necessary components for network virtualization and network security.

The components of the Project include: an SDN controller, virtual router, analytics engine, published northbound APIs, hardware integration features, cloud orchestration software and an extensive REST API.



Structure of TF Community

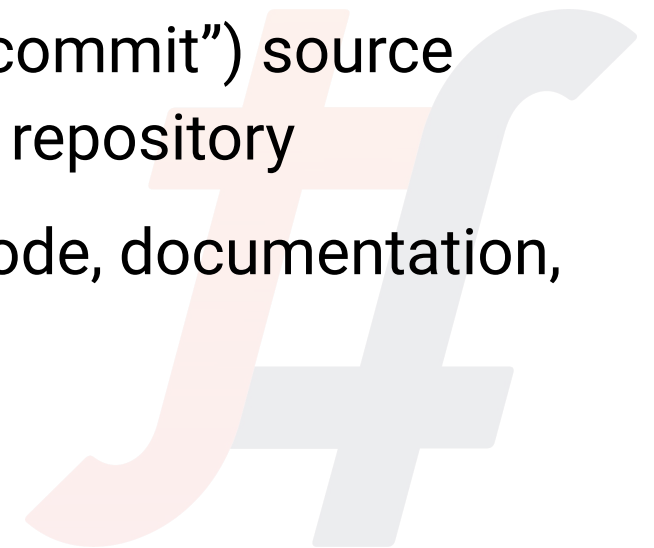
Technical Steering Committee - main body responsible for setting high level architecture goals and coordinating overall project architecture and technical direction

Release Manager - elected for a one year cadence TF member responsible for setting and monitoring release process

Project Technical Leads - provide direction for a sub-project

Committers - Contributors who earned the ability to modify (“commit”) source code, documentation or other technical artifacts in a project’s repository

Contributors - anyone in the TF community that contributes code, documentation, or other technical artifacts to the Project



Modules and project structure

- Tungsten Fabric functionality is logically separated into modules¹ representing the relevant functional area:
 - Configuration System
 - Fabric Management
 - Control Plane
 - Data Plane
 - Kubernetes Orchestrator
 - Openstack Orchestrator
 - Deployment
 - Packaging
 - Analytics
 - WebUI
- Sub-projects² are collections of work undertaken to deliver a well-defined goal. They transcend modules and component:
 - TF Core
 - TF Operator
 - CI/CD
 - Documentation & Training

1. <https://wiki.tungsten.io/x/PoRdAQ>

2. <https://wiki.tungsten.io/x/6AELAQ>



Release process principles

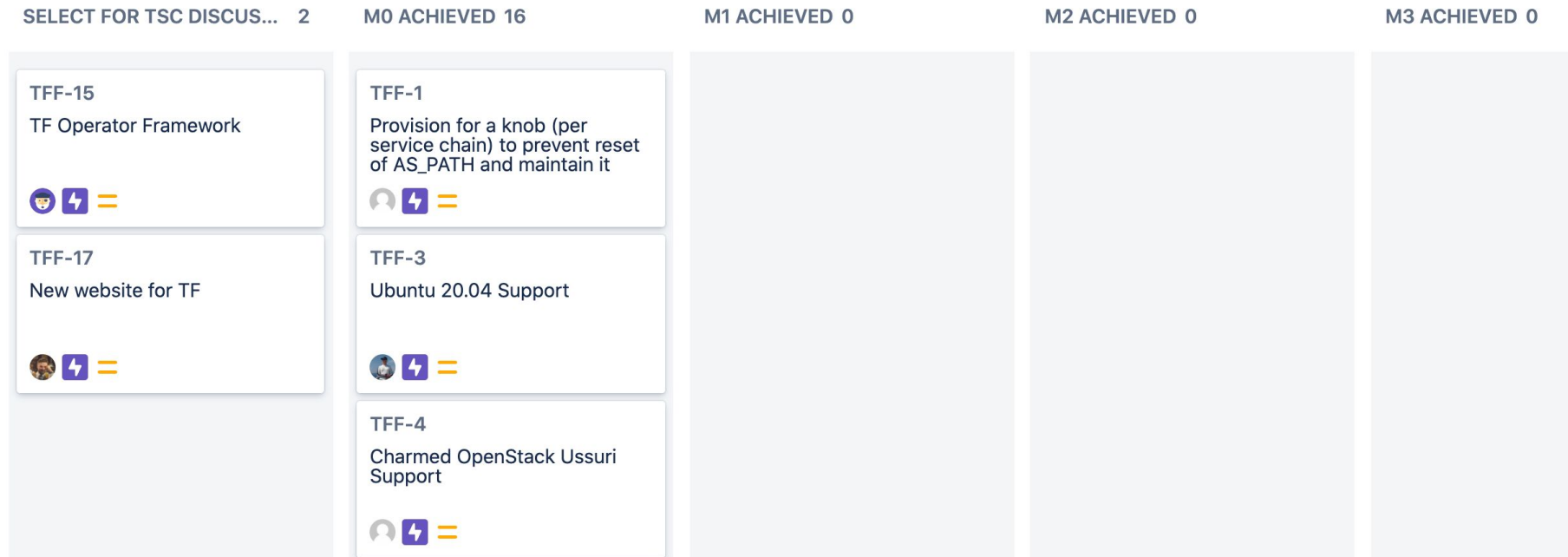
Milestone	Goal	Description
M0	Declaration of participation in the release, initial scope of release	<ul style="list-style-type: none">• Blueprint proposal represented as a Jira EPIC ticket / Feature lead assigned to Jira Epic ticket• The initial version of Increment Blueprint stored on github/tf-spec/Release folder,• The blueprint presented, discussed and approved by related approvers• Release Scope Overview updated accordingly, agreed, approved by TSC• Dependencies between projects/components/areas identified
M1	Dependencies between modules, new functionalities, release plan	<ul style="list-style-type: none">• All dependencies between existing functionality and requested feature discussed and agreed by PTL and related approvers• Jira Epic ticket broken down into stories in related Jira project (backlog, plan) - presented and discussed on TWS meeting• The final version of Blueprints, Release Scope and Release Plans discussed and approved by approvers
M2	Feature/Functionality technical design freeze	<ul style="list-style-type: none">• Technical design fully provided, agreed (documentation on the Confluence/github/tf-spec available)• Jira Epic ticket updated with link to technical design• Documentation started / Feature tests started• Feature/Functionality Freeze - Technical Design approved by relevant approvers
M3	API freeze (beta available)	<ul style="list-style-type: none">• External API available for beta-tests• Jira Epic ticket updated with information about API availability (documentation link - tf repo)• Documentation in progress / Tests in progress• API Freeze confirmed by Feature Lead (comment on Jira Epic ticket with repo/commit ID link)
M4	Code freeze - only bug fixing allowed	<ul style="list-style-type: none">• All functionality and APIs available for testing• Jira Epic ticket updated with information about final API documentation, executed UI tests documentation• Documentation provided - confirmed by Documentation project PTL• Code freeze confirmed by Feature Lead (comment on Jira Epic ticket) - only bug fixing allowed
RC0-3	Release candidates freeze - branch stabilization	<ul style="list-style-type: none">• Release candidates agreed, approved, tested• System tests conducted, quality confirmed by CI/CD PTL• Final documentation provided, reviewed, approved by Documentation PTL• Marketing information provided to Marketing Advisory Council• Release branch cut off - branch stabilization from now on• Release Candidates freeze, confirmed by PTL and TSC
Deployment	Release deployment	<ul style="list-style-type: none">• Release deployment

Release process principles

Release TFF board

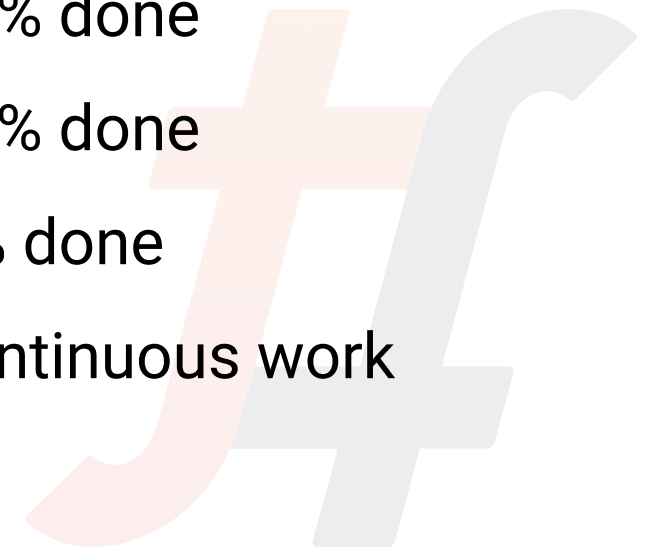
Kanban board

QUICK FILTERS: [Only My Issues](#) [Recently Updated](#)



Documentation process

- Define areas that documentation should cover - 90% done
- Catalog existing documents - 75% done
- Create JIRA tasks to track progress - 100% done
- Establish rules and requirements for documenting - 0% done
- Transfer documentation from Juniper - 25% done
- Define what is missing - 25% done
- Fill gaps - 0% done
- Ensure maintenance of documentation - continuous work



Areas of documentation

- Governance
- Release process
- Support activities (TF CI/CD, website maintenance etc.)
- Installation
- Deployment
- Development
- Contribution / Community



Documentation

- Wiki: wiki.tungsten.io/Documentation
- JIRA: <https://jira.tungsten.io/projects/DOC>



How to join TF Community

1. **Create an account** at Linux Foundation
myprofile.linuxfoundation.org
2. **Login** to jira.tungsten.io and wiki.tungsten.io so your **account will be populated** to those systems
3. **Go to** tungsten.io/community/ to see community **meetings calendar** and learn about **different ways** you can connect with the community
4. **Join** community meetings, slack channels and mailing lists
5. **Start contributing**



Join us!

LF NETWORKING
Virtual Technical Meetings



Marek Chwal

Release Manager

Tungsten Fabric, **codilime**
CREATING VALUE

marek.chwal@codilime.com



Szymon V. Gołębiewski

Documentation & Training PTL

Tungsten Fabric, **codilime**
CREATING VALUE

szymon.golebiewski@codilime.com

