



LFN Developer & Testing Forum

# **Tungsten Fabric Release Process & Planning**

**Marek Chwal, James Kelly**

# Agenda

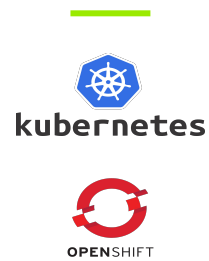
- Structure of TF Community
- Project structure
- Release process principles
- R21.05 scope
- Next Release scope
- How to become a TF member



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



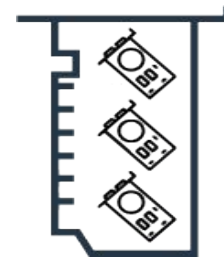
orchestrators or  
virtualization  
platforms



smartNIC  
devices



SR-IOV  
clusters



Cloud Platforms



3rd party  
solutions



# Structure of TF Community

## Technical Steering Committee

- elected from active community members
- consist of 6 seats
- fostering cross-project collaboration
- defining technical best practices and community norms
- responsible for setting high level architecture goals
- coordinating technical community engagement

## Release Manager

- responsible for setting and monitoring release process
- coordinating overall transparency of performed work



# Structure of TF Community

## Project Technical Leaders

- provide direction for sub-project
- spokesperson for the project
- responsible for transparency of the project work

## Feature Leads

- lead the work within the epic
- responsible for the transparency of epic artifacts





# Structure of TF Community

## Committers

- good knowledge about module
- ability to modify (“commit”) source code, documentation or other artifacts  
decision maker on all matters for a project (design, code, patches)
- rights per repository/module

## Contributors

- contributes to a project
- work with a project’s committers
- contributions could take the form of code, code reviews, or other artifacts



# Project structure

Tungsten Fabric sub-projects<sup>1</sup>:

- **TF Core** - umbrella project, covers almost all modules<sup>2</sup>
- **CI/CD** - continuous integration/delivery tasks
- **Documentation & Training** - materials, release notes, etc
- **TF Operator** - well defined separated scope of the product

1. <https://wiki.tungsten.io/x/6AELAQ>  
2. <https://wiki.tungsten.io/x/0oVdAQ>



# Release process - why & what for?

- Plan and schedule
- Structured and standardized way of presenting partial results and final output
- Checklist of expectations
- Transparency

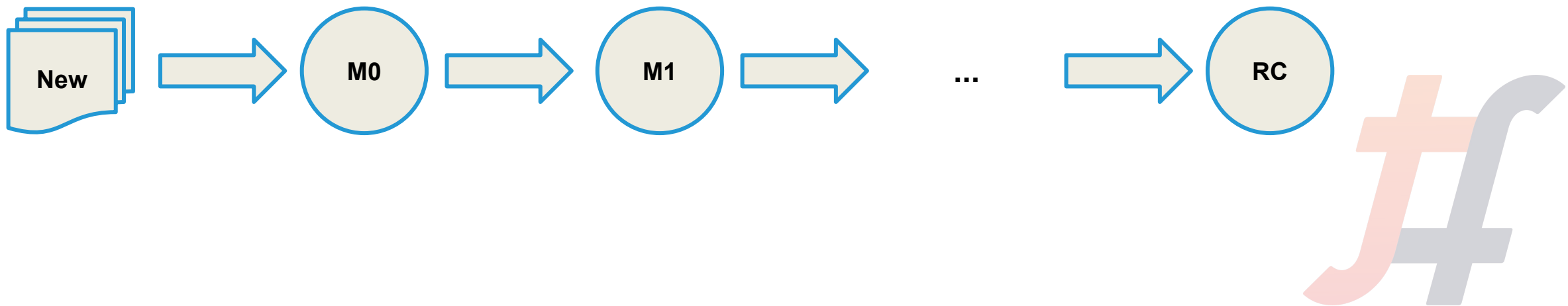
**Release process defines the Community!**





# Release milestones

- Acceptance criteria for a well-defined piece of work
- Checklist for each stage
- Clear expectation of partial result on a particular stage



# Release Process - M0

- Declaration of participation in the simultaneous release
- Blueprint proposal represented as a Jira EPIC ticket
- Feature lead assigned to Jira Epic ticket
- The initial version of Increment Blueprint stored on github release folder
- The blueprint presented, discussed and approved by related committers
- Release Scope Overview updated accordingly
- Dependencies between projects/components/areas identified

**Declaration of participation in the release, initial scope of release**



# Release Process - M1

- All dependencies between existing functionality and requested change/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 for participating projects discussed and approved by approvers

**Dependencies between modules, new functionalities, release plan**



# Release Process - M2

- 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 (Modules/Committers)

**Feature/Functionality technical design freeze**



# Release Process - M3

- 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)

**API freeze (beta available)**



# Release Process - M4

- 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

**Code freeze - only bug fixing allowed**





# Release Process - RC

- 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 PTLs and TSC

**Release candidates freeze - branch stabilization**



# Release Process - Deployment

- Tested Release branch
- Quality confirmed
- Known errors list prepared
- Release notes prepared
- Date of official publish agreed and approved by PTLs and TSC

**Official Release Issued**



# Important links

- [Wiki of Tungsten Fabric release process](#)
- [Tungsten Fabric jira](#)
- [General writing style guidelines](#)
- [Cheat sheet for reStructuredText](#)



# TF Release process

## Release Process Proposals

Utworzone przez Casey Cain, ostatnia modyfikacja przez Marek Chwal przed momentem

- [The purpose of the release process](#)
- [Prerequisites](#)
- [Release Process](#)
  - [Milestones goal and acceptance criteria](#)
- [Release Cadence, Life Time & Support](#)

### The purpose of the release process

The Release Process is dedicated to creating a plan and then following and monitoring that plan to deliver an increment to the product. It is extremely important, especially in a community where there are lots of developers from different companies, countries, to have a common understanding of what and how we would like to achieve. The release plan will help us to coordinate our efforts to achieve agreed goals and to deliver proper, coded, tested, and working as an expected increment of product. To achieve that, we need to act in accordance with agreed rules, time schedule, and milestones.

### Prerequisites

All who would like to attend community effort and contribution must follow bellow prerequisites:

- [Read the Getting Started as a Contributor page](#) (including Registration and User Creation, environment installation)
- [Check/assign to one of the existing projects](#) or create a [proposal for a new project](#)

### Release Process

The unit piece of functionality/work which can be included in the release is represented by the Epic ticket in Jira. The release process consists of several Milestones in the pipeline of epics with clearly defined acceptance criteria. One can treat these acceptance criteria as a checklist that has to be followed to say that particular epic achieved this Milestone. Overall flow is outlined as follows:











1. [Invitation of proposals for features, enhancement requests](#) - approved till M0
2. [Blueprints/Design/Architectural approval cut-off date](#) (4 months before release date)
3. [Code delivery cut-off date](#) (2 months before release date) - till M4
4. [Branch cut-off date](#) (1 month before release date)

# Release process in Jira

Release TFF board

## Kanban board

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


| SELECT FOR TSC DISCUS... 2  | M0 ACHIEVED 16   | M1 ACHIEVED 0 | M2 ACHIEVED 0 | M3 ACHIEVED 0 |
|---|--|---------------|---------------|---------------|
| <p>TFF-15<br/>TF Operator Framework</p> <p>  =</p> | <p>TFF-1<br/>Provision for a knob (per service chain) to prevent reset of AS_PATH and maintain it</p> <p>  =</p> |               |               |               |
| <p>TFF-17<br/>New website for TF</p> <p>  =</p>    | <p>TFF-3<br/>Ubuntu 20.04 Support</p> <p>  =</p>   |               |               |               |
|   | <p>TFF-4<br/>Charmed OpenStack Ussuri Support</p> <p>  =</p>   |               |               |               |

Source: <https://jira.tungsten.io/secure/RapidBoard.jspa?rapidView=5&projectKey=TFF>







# Release process in Jira

 TF Features / TFF-22

## Integration of Intel PAC N3000

[Edit](#) [Comment](#) [Assign](#) [More](#) [Backlog](#) [Withdrawn](#) [MO achieved](#) [Admin](#)

**Details**


|                    |  |                |  |
|--------------------|--|----------------|--|
| Type:              |  Epic   | Status:        | <b>SELECTED FOR DIS...</b> <a href="#">(View Workflow)</a> |
| Priority:          |  Medium | Resolution:    | Unresolved   |
| Affects Version/s: | None   | Fix Version/s: | <a href="#">R21.12</a>                                     |
| Labels:            | None   |                |  |
| Epic Name:         | Integration of Intel PAC N3000   |                |  |

**Description**


TF vRouter-DPDK integration with Intel PAC N3000 board


- Blueprint document submitted for review: <https://gerrit.tungsten.io/r/c/tungstenfabric/tf-specs/+64983>


**Attachments**


 Drop files to attach, or [browse](#).

**People**

Assignee:  Epic Leader Name  
[Assign to me](#)

Reporter:  Reporter Name

Votes:  [Vote for this issue](#)

Watchers:  [Start watching this issue](#)

**Dates**

Created: 15/Dec/20 2:32 AM

Updated: 01/Jun/21 1:33 PM

**Agile**

[View on Board](#)





# Current Release 2021.05

- Kubernetes 1.18/19
- KubeVirt support
- Red Hat OpenShift 4.5/4.6
- Red Hat OpenStack 16.1 / RHEL 8.2
- Canonical Ussuri OpenStack and Ubuntu 20.04
- Data plane MAC learning
- DPDK upgrade to v19.11
- <https://docs.tungsten.io/en/latest/user/index.html> (to be updated post release)



# Next Release scope

- Red Hat OpenStack 16.2 / RHEL 8.4 and latest RHOSP 16.1.z/RHEL 8.2.z
- Operator Framework of lifecycle management of TF on Kubernetes
- Intel N3000 SmartNIC
- vRouter L3 Multihoming
- vRouter Load Balancing next-hop stickiness in Kubernetes LB scenarios
- Security policy across Kubernetes clusters




# How to join TF Community

1. **Create an account** at Linux Foundation  
[myprofile.linuxfoundation.org](https://myprofile.linuxfoundation.org)
2. **Login** to [jira.tungsten.io](https://jira.tungsten.io) and [wiki.tungsten.io](https://wiki.tungsten.io) so your **account will be populated** to those systems
3. **Go to** [tungsten.io/community/](https://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!



**Marek Chwal**  
Release Manager  
Tungsten Fabric,  **codilime**  
CREATING VALUE

[marek.chwal@codilime.com](mailto:marek.chwal@codilime.com)



**James Kelly**  
PLM, Cloud-native architect  
Tungsten Fabric,  **JUNIPER**  
NETWORKS

[jamesk@juniper.net](mailto:jamesk@juniper.net)