

Tungsten Fabric Deprecation Policy (DRAFT)

Purpose:

Currently, it appears the best resource to understand feature deprecation comes from Juniper's release notes.

https://www.juniper.net/documentation/en_US/contrail5.0/information-products/topic-collections/release-notes/jd0e562.html

However, release notes are only accessible once the release is published, so there is opportunity for confusion for new or existing community members.

Example:

Release 5.1 has a blueprint illustrating a plan to support Mesos integration. However, 5.1 also announced "deprecation" for Mesos support.

<https://github.com/Juniper/contrail-specs/blob/master/5.1/contrail-mesosphere-integration.md>

Similarly, it may not be clear why or when Tungsten Fabric APIs get deprecated. The VNC API documents specific objects or portions of the schema that are deprecated. For other APIs such as the router agent, it may be unclear why or when the API changes. It looks like Jira tickets and Blueprints capture some of these deprecations, but this may not be a comprehensive exercise.

http://www.opencontrail.org/documentation/api/r3.2/contrail_openapi.html

<https://github.com/Juniper/contrail-controller/wiki/Blueprint-Format>

The purpose of this wiki page is to outline a community approved deprecation policy that will evolve with our community to ensure transparency and communication with current and prospective community members.

Goals:

Start documenting a deprecation policy. As a community, we should expect it to evolve and improve over time. That said if we start now and avoid disrupting current development, it might help avoid barriers to using Tungsten Fabric. Also, this could help get more TSC involvement with developers to ensure changes to APIs and features are documented and communicated. Lastly, if we want new developers from other organizations, we will need documentation like this to level set across the community.

Short Term Ideas:

- All deprecated features **must** be documented in public Tungsten Fabric announcements (blogs) and release notes for every release. This shall be owned by the TSC. Other CNIs approach this similarly.
 - <https://www.projectcalico.org/whats-new-in-calico-v3-3/>
- API deprecations **should** be documented in similar public Tungsten Fabric announcements (blogs) and release notes for every release. This shall be owned by the TSC.
- In turn, the TSC **should** drive updates to internal collateral accordingly. This shall be owned by the TSC.
 - Developer guides
 - Powerpoint collateral
 - Others?
- Check if this verbiage already exists in the OpenContrail repositories and update/align as needed.
- In summary, this is status quo, but documenting the policy can ensure we comply accordingly.

Longer Term Ideas:

- Commit to using Jira tickets (or BluePrints) to document deprecated features or APIs. Open for discussion.
 - It looks like this is roughly in place already:
 - <https://jira.tungsten.io/browse/TFB-1269>
 - Also, a portion of deprecations are supposed to be captured in Blueprints.
 - <https://github.com/Juniper/contrail-controller/wiki/Blueprint-Format>
 - Maybe a quick audit may be in order to ensure this is enforced.
- Investigate using a 'deprecation period' during which features or APIs must work after announcement of deprecation.
- Align all API documentation accordingly.
 - Example:
 - <https://developers.evrthng.com/docs/deprecation>

Proposed Verbiage for GitHub/Gerrit:

Tungsten Fabric Feature Deprecation Policy

This document outlines the policies that guide feature and API deprecation within the Tungsten Fabric community. The goal of Tungsten Fabric is to provide the "network fabric" for any cloud and any orchestrator. That said, like any open source community, we must prioritize community needs which may require deprecating features or APIs. It is important for all community members to understand the policies that guide how Tungsten Fabric communicated these changes with others within the community.

Feature and API Deprecation:

The technical steering committee is responsible for communicating all deprecated features and APIs in release notes and using public announcements. See the following for example [release notes](TBD) and [announcements](#).

Our goal is to ensure all members understand how Tungsten Fabric changes when it is released. That said, at this time, we do not have a policy that outlines how long deprecated features or APIs shall be supported. We are working on updating the release cycle, so as that progresses, we shall update the policies mentioned in this document.

Issues and Exceptions:

We recognize that we may miss some considerations of community members in this policy. If so, there are a number of ways to engage us to give feedback or if exceptions are required. We want all community members to be successful, so here are some ways to engage us to check in on a change including mailing lists and Jira tickets.