Modules and Components Proposal

Modules

Tungsten Fabric functionality is logically separated into Modules representing the relavant functional area.

Components

Components are recognized as parts that make up modules or other components - they may be reused across other components and modules. Modules may be composed of components

Repositories

Modules and components may or may not align with source code repositories, a module may span across multiple repositories or in cases a repository may host code for multiple modules

Committer

For each project/module there is a set of Contributors approved for the right to commit code to the source code management system (the “Committers”) for that project/module.

- Committer rights are per module; being a Committer on one module does not give an individual committer right on any other module.
- The Committers will be the decision-makers on all matters for a module including implementation, code, patches.
- Committers are the best available individuals but usually work full-time on modules/components in active development.
- All project/module committers information such as name, company, and Contact information should be documented in the wiki under the project.

In order to preserve meritocracy in the selection of Committers while ensuring diversity of Committers, each initial project is encouraged to take on at least two Committers from different companies (subject to meritocracy).

The following table defines all the modules adopted under Tungsten Fabric, along with additional information on

- Components it includes
- Repositories where the code spans
- Committers for the module
- Short Description

Module	Components	Repositories	Committer	Short Description
Configuration System	Schema Transformer, Cassandra, Zookeeper, API Server, Service Monitor, RabittMQ	tf-common tf-controller	Nagendra Prasath Maynattamai Prem Chandran Ignatious Johnson christopher	A set of software services and APIs for maintaining and managing virtual network configurations as expressed in 'intent' configuration form by clients of Tungsten Fabric. Said intent is persisted and compiled into a granular form distributed and consumable by down steam services such as Control plane and Fabric Management modules. Cassandra and Zookeeper are used for persistence, API server for client interaction, Schema Transformer for intent compilation.
Fabric Management	Device Manager, Overlay, Underlay, Device Configuration, Device Manager Infrastructure, Image Management	tf-common tf-controller	Atul Moghe	A system for management for physical fabric devices.
Control Plane	Control Node, vrouter Agent	tf-common tf-controller	Mahesh Sivakumar Pranavadatta DN sangarshan pillareddy	A set of software services that maintain a model of networks and network policies, typically running on several servers for high availability. The Tungsten Fabric controller integrates with cloud management systems such as OpenStack or Kubernetes. Its function is to ensure that when a virtual machine (VM) or container is created, it is provided with network connectivity according to the network and security policies specified in the controller or orchestrator.
Data Plane	vrouter, encryption, DPDK, Kernel, SmartNIC, SRIOV	tf-vrouter	Anand Rao Kiran KN	The default deployment option today is for the vRouter forwarder to be implemented in a module that runs in the Linux kernel. The vRouter implements networking functionality that would otherwise be performed using iptables or Open vSwitch. Running in the kernel gives the forwarder direct access to network traffic as it passes through the network stack of KVM, and provides a significant performance improvement over what can be achieved if the forwarder ran as a process in userspace.
Kubernetes Orchestrator	Openshift	tf-controller	Prasad Miriyala Sachchidanand Vaidya	Support for OpenShift integration with Tungsten Fabric.

Openstack Orchestrator	Neutron Plugin	tf-controller	Nagendra Prasath Maynattamai Prem Chandran	Support Neutron plugin integration with Openstack and allow Tungsten Fabric to operate as a network provider for OpenStack.

Deployment	Juju, Ansible, Helm, Kolla, Meso, OpenShift, RHOSP	tf-openshift-ansible
		tf-helm-deployer
		tf-charms
		tf-ansible-deployer
		tf-kolla-ansible
			Alexandre Levine	Containers and deployers layers for TF. Includes:
				• Containers
				  ◦ Docker
				  ◦ Podman
				  ◦ Cri-o
				  ◦ Containerd
				• Deployers
				  ◦ RHOSP (13, 16) - openstack, vrouter-less / RHEL 7, 8
				  ◦ JuJu - openstack, k8s, hybrid / CentOS 7, Ubuntu 16, 18
				  ◦ Ansible deployer - openstack, k8s / Centos 7, Ubuntu 16, 18
				  ◦ Helm - openstack, k8s / Centos 7, Ubuntu 16, 18
				  ◦ k8s manifests - k8s / Centos 7, Ubuntu 18
				  ◦ Openshift (3, 4) - RHEL 7, 8
				  ◦ Future: Operator framework
				• Integration
				  ◦ Akraino / Airship
				  ◦ OPNFV

Packaging	tf-container-builder tf-packages	tf-packages tf-container-builder tf-deployers-containers tf-build	Alexandre Levine	Tungsten Fabric build and packaging as RPMs and container images
Operator Framework		tf-operator	Michael Henkel Sanju Abraham Prabhjot Singh Sethi	tf-operator to manage lifecycle of tungsten Fabric deployments in kubernetes native deployments
Analytics		tf-common tf-analytics	Sachchidanand Vaidya	
WebUI	NodeJS Server	tf-web-core tf-webui-third-party tf-web-controller	@zsong • Manoj Naik • Biswajit Mandal	
Controller Test Framework		tf-test	TBD	

Note:- Technical committee members will be automatically nominated as committers for modules with missing(/TBD) committers