



LFN Developer & Testing Forum

# TF deployers and CI

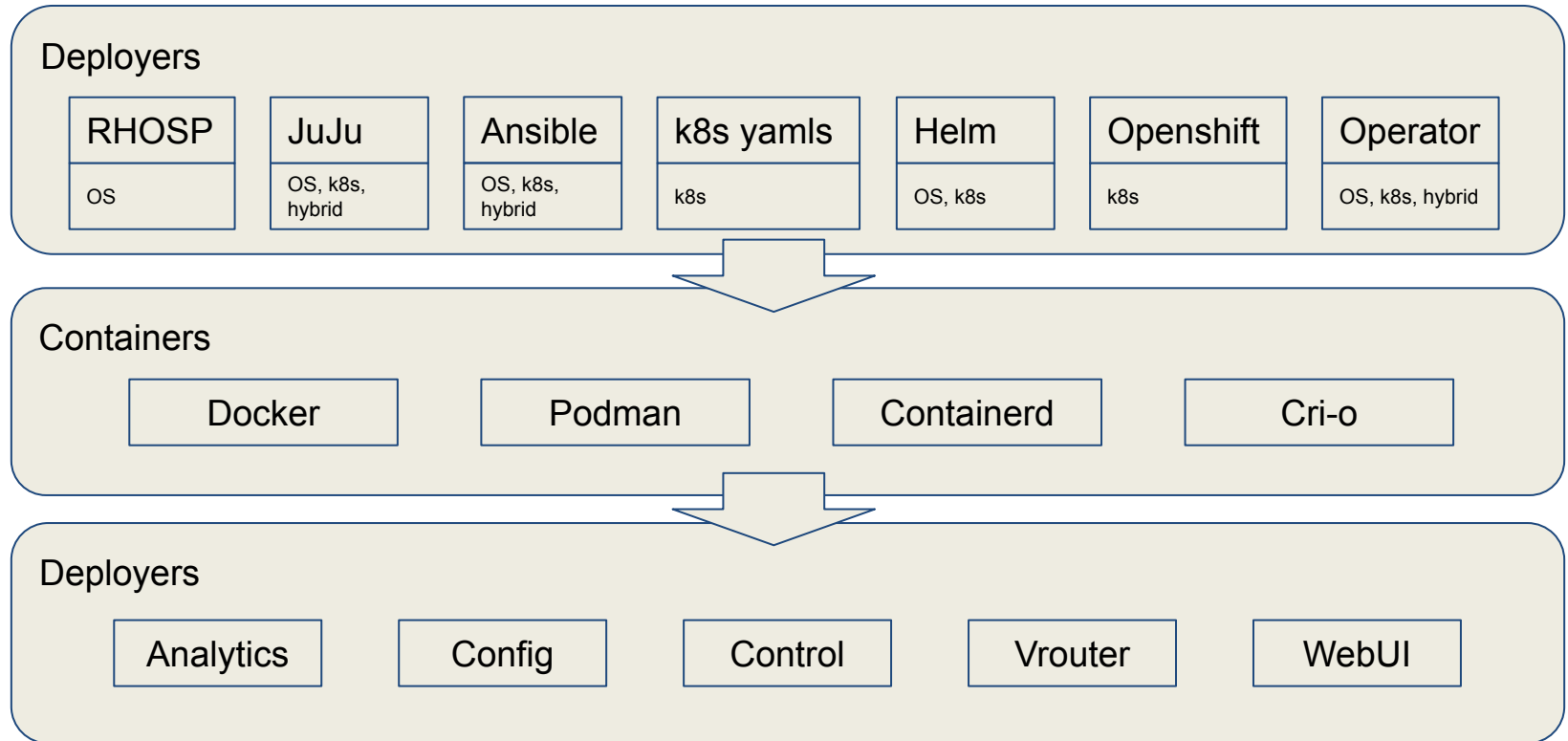
@twitterhandle

# Purpose of the presentation

- The presentation's goals are:
  - Give overview of TF deployment methods
  - Describe tactics and instruments employed to develop and maintain the variety of deployers and configurations



# Architecture and variety



# Problems addressed

- Variety to support
  - A lot of deployment methods, configurations to develop and maintain
- Slow and narrow CI
  - Original CI did checking with ansible deployment only, took 4-8 hours for review
- Hard to install
  - Various deployers were differently hard to deploy, but all were hard and manual
- Complicated development
  - Hard to build, deploy, and test manually - CI later did it differently and produced other errors

# Dimensions

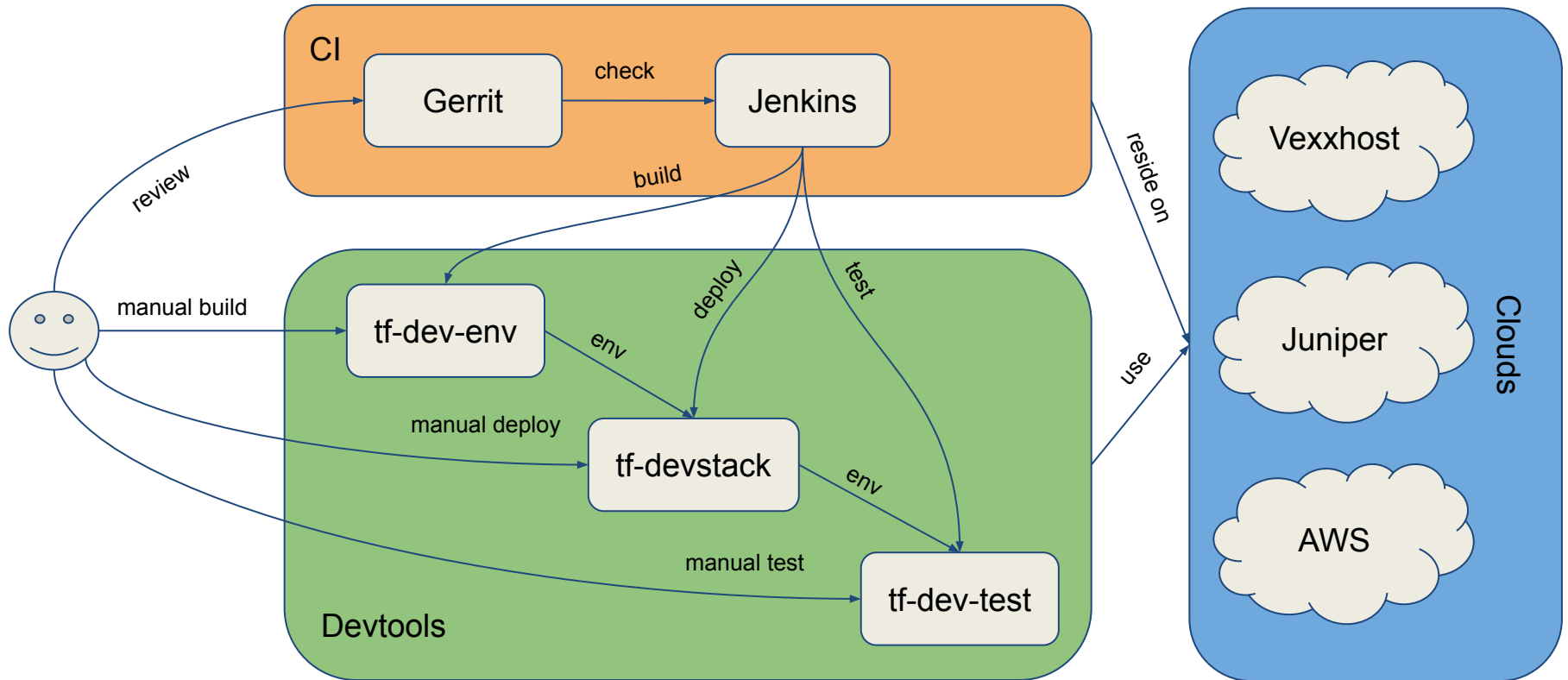
- Deployer - RHOSP 13/16, JuJu, Ansible, Helm, k8s yaml, Openshift 3/4, Operator, RHOSP-ML2, bare containers
- Configuration - All-in-one, HA, Production (bare-metal)
- Orchestrator - OpenStack, k8s, Hybrid
- OpenStack - Ocata, Queens, Rocky, Train, Ussuri
- TF release - iLTS, latest, non-iLTS support
- Cloud - bare-metal, Openstack (various), AWS, GCP, Azure
- Linux - Ubuntu 16.04/18.04/20.04, CentOS 7/8, RHEL 7/8
- Modes - Kernel, DPDK, ML2, SRIOV

# Tactics used

- Automation
  - Automate and make reproducible all of the processes and infrastructure
- Thorough and faster CI
  - Cut build/check/gate processes timings and add more various configurations for automated checks
- Single-click universal installer
  - Create universal installer for development and testing purposes with easy and robust deployment for major dev/test configurations
- Unification of CI and development tools
  - Use the same devtools to allow manual build/deploy/test and for CI automated processes.



# Infrastructure



# CI/QA flow

## Package build

TF is built.  
Deployers source code is packaged into containers.

## Deploy TF

TF is deployed in various configurations during:

- Manual devtools testing
- CI checking
- CI gating
- CI nightly

Contrail status is checked.

## Run sanity test suite

TF is checked with “Contrail sanity” test suite.

## Run deployer test suite

TF is checked with specific deployment tests.  
Right now this suite has started to appear and contains ZIU-related tests.

## Run manual tests

TF is checked manually for configurations and/or features not yet automated. Mostly concerns with RHOSP, upgrades, and external deps like importing cluster to Contrail Command.



1. Unit-tests
  - run by tf-dev-env, used in CI, check particular unit operability
2. Contrail CI Sanity test suite
  - run by tf-dev-test, used in CI, check whole solution
3. Deployment tests
  - run by tf-dev-test, used in CI, specific tests which check various intricate process (like ZIU) or particular cases, they get filled in with tests for new features or for checking problems reported by customers
4. Smoke tests
  - run by tf-dev-test, quick validity checking tests, not fully complete test suite
5. Contrail Extended Sanity & other test suites
  - run by QA team, provide more thorough functionality and performance testing

Testing in TF CI occurs in three stages:

## 1. Checking

- pre-approval testing, least amount of configurations tested for fast response

## 2. Gating

- post-approval testing, sufficient amount of configurations tested to allow code merging

## 3. Nightly

- pre-publish build testing, extended amount of configurations, including baremetal, tested to allow build publishing and marking it stable

# Deployment automation matrix

Types	RHOSP*	JuJu	Ansible	Helm	k8s yaml	Openshift*	Operator
AIO	O	OKH	OK	OKH	K	K/wip	wip
HA	O	OKH	OK	OKH	K	K/wip	wip
Prod (baremetal)	O	O					tbd
iLTS	O	O					tbd
ZIU	O	O	O				tbd
ISSU	O	O	O				tbd
DPDK/Sriov	O	O					tbd

O - Openstack, K - Kubernetes, H - Hybrid, **Orange - Automated**

\* - Several versions - RHOSP 13/16, Openshift 3/4

- Gerrit (review)
  - <https://gerrit.tungsten.io/>
- Jenkins (check)
  - <https://tf-jenkins.progmaticlab.com/>
- Matrix Status (monitor)
  - <http://tf-monitoring.progmaticlab.com/>
- tf-dev-env (build)
  - <https://github.com/tungstenfabric/tf-dev-env/>
- tf-devstack (deploy)
  - <https://github.com/tungstenfabric/tf-devstack/>
- tf-dev-test (test)
  - <https://github.com/tungstenfabric/tf-dev-test/>

A high-quality space photograph of Earth from orbit, showing the curvature of the planet, blue oceans, green landmasses, and white clouds. A bright sun is visible on the left horizon, creating a lens flare effect. The background is a deep black space filled with stars.

# OLF NETWORKING

---

LFN Developer & Testing Forum