



Integration Lab

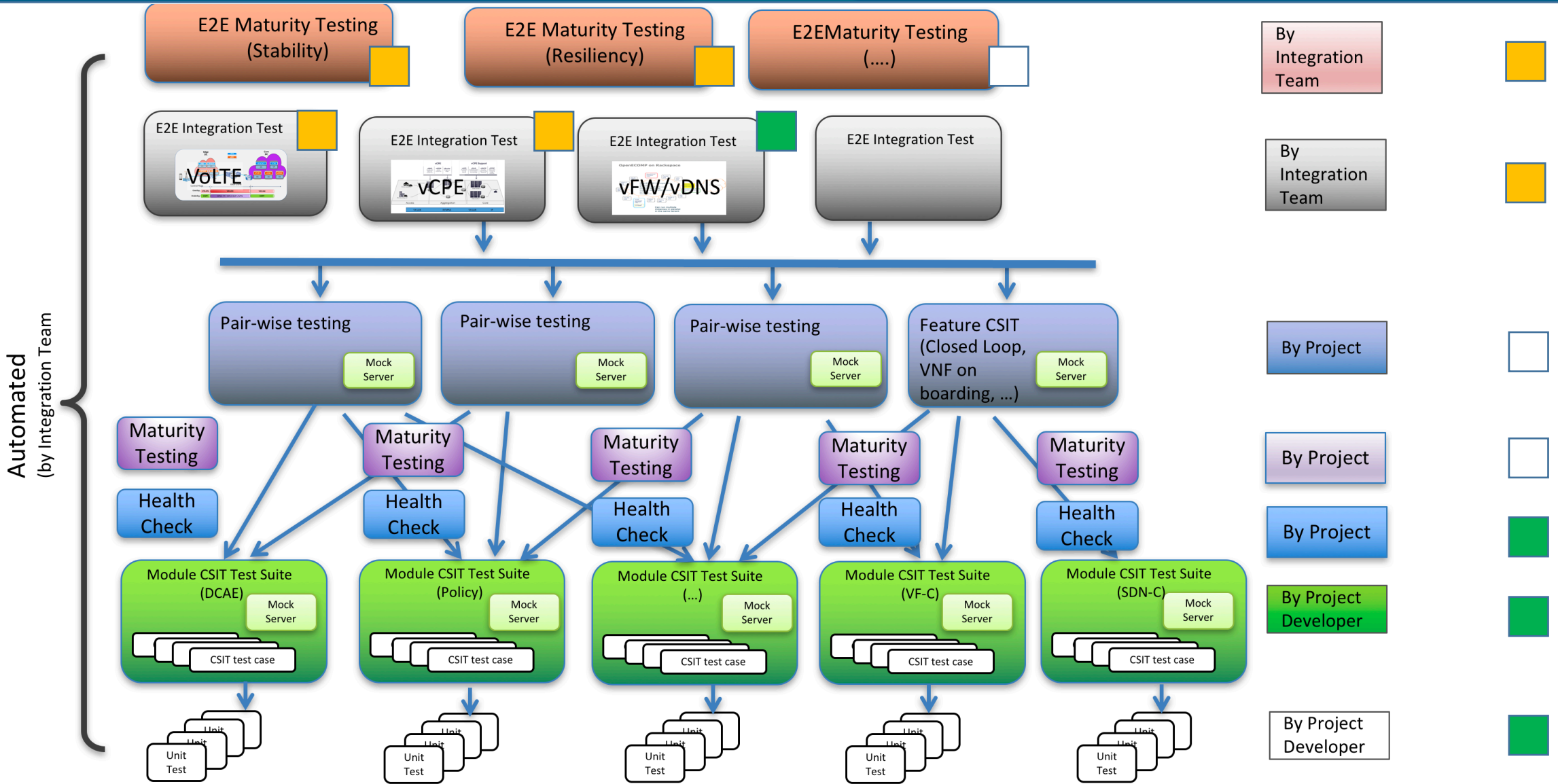
- How to leverage Integration Lab and Related Tools for CI?

Helen Chen, Stephen Gooch, Scott Blandford, Sylvain Desbureaux, Gary Wu

Agenda

- **ONAP CI Journey**
- **How to access Integration Labs?**
- **ONAP Integration Tools**

Our CI Journey



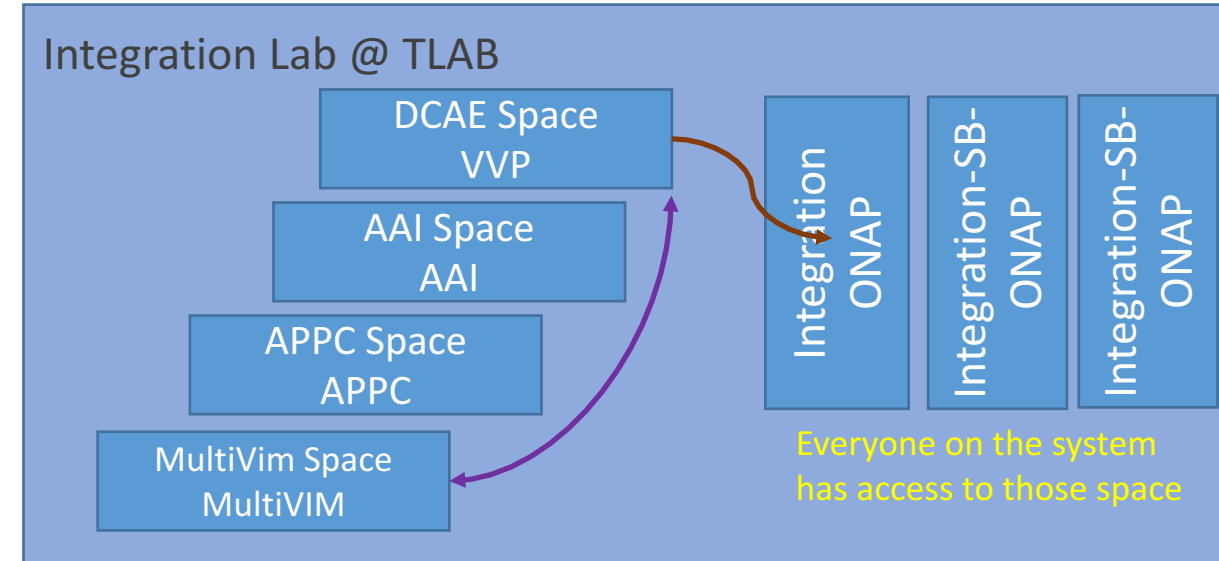
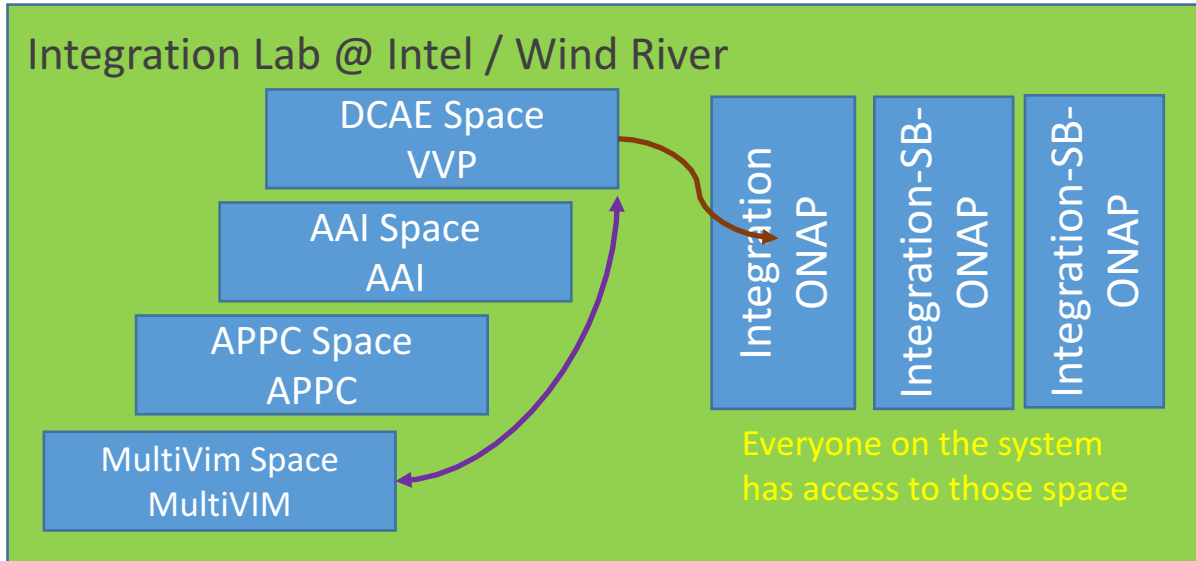
Our goal is to automate all the testing to achieve continuous integration / release

Full ONAP Beijing Release Deployment Recommendation

- Supported Method: HEAT and OOM
- Resource usage:
 - HEAT: xlarge (16/160)*6 + large (8/80)*7 + medium (4/40)*5 + small (2/20) + 100 disk volume = 176 GB RAM + 1860 GB disk
 - OOM: No clustering:
 - Rancher: 8 GB RAM + 80 GB Disk
 - 9 Nodes: 16 GB RAM + 160 GB disk
 - Total: 152 GB RAM, 1520 GB Disk
 - OOM with Clustering: (with 3 APPC nodes and 3 SDNC nodes)
 - Rancher: 8 GB RAM + 80 GB Disk
 - 12 Nodes: 16 GB RAM + 160 GB disk
 - Total: 200 GB RAM, 2000 GB Disk (recommended)

Need do more optimization at Casablanca

Integration Lab Deployment Diagram



- **Access**
 - We will support both OOM and Heat deployment
 - Each project has your own tenant
 - Each project has access to all Integration's tenants
- **Tools:** we'll install all necessary ONAP Maturity testing related tools and simulator in Integration tenants space.



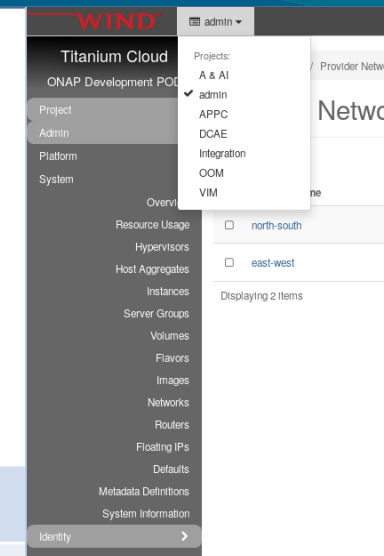
Intel HF2 ONAP Community POD 01 (AKA: POD-25)

Stephen Gooch, Wind River, stephen.gooch@windriver.com

ONAP Developer POD - Projects

- **Integration** – every has visibility.
- **Individual Projects**

A & AI	AAF	APPC	CLAMP	CC-SDK
DCAE	DMaaP	External-API-Framework	Holmes	Logging
Microservices	Modeling	VIM	CLI	PFPP
PPPP	SDN-C	SO	VID	VFC
VNF-SDK	OOM	SDC	PAF-PAL	vCPE
Integration-SB-00	Integration-SB-01	Integration-SB-02	Integration-SB-03	



How to Request Access to Wind River Lab?

- Open JIRA: <https://jira.onap.org/projects/OPENLABS>
 - Component MULTI_GEO LAB
 - Which project you belongs to?
 - Assign to Stephen Gooch

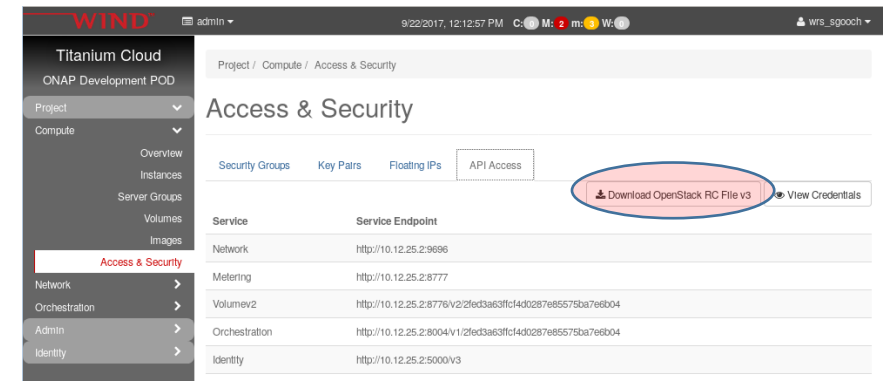
ONAP Developers Lab – Using the VPN

- From the JIRA, a form letter is sent with an encrypted zip file.
 - There are two files in this zip
 - login.txt (your username and password)
 - pod-onap-01.ovpn (Open VPN CA same for all)
- Edit pod-onap-01.ovpn to include login information
 - Linux/Mac
 - auth-user-pass login.txt
 - Windows
 - auth-user-pass “C:\\Program Files\\OpenVPN\\config\\login.txt”
 - You may need to add “http-proxy [url:port](#)” or “socks-proxy [url:port](#)”
 - Discuss with your IT representative.
- Execute
 - Linux
 - `$ sudo openvpn --config pod-onap-01.ovpn`
 - Windows
 - Install the Open VPN client, import pod-onap-01.ovpn and select connect

ONAP Developers Lab – Using remote CLI

- After connecting the VM. You can use a local Linux machine or the supplied jumpstation for OpenStack CLI access.
- Download OpenStack RC File from UI:
Project -> Compute -> Access & Security -> API Access
- Upload file to jumpstation (or local machine)
 - If running on local machine, ask stephen.gooch@windriver.com for remote CLI SDK.
- Source and run OpenStack commands

```
user@pod-onap-01-vjhost:~$ source ./Integration-openrc.sh
Please enter a path for your CA certificate pem file, #or press enter if you are not using HTTPS
Please enter your OpenStack Password for project Integration as user username:
user@pod-onap-01-vjhost:~$ openstack usage list
```



- Jumpstation IP 10.12.5.50

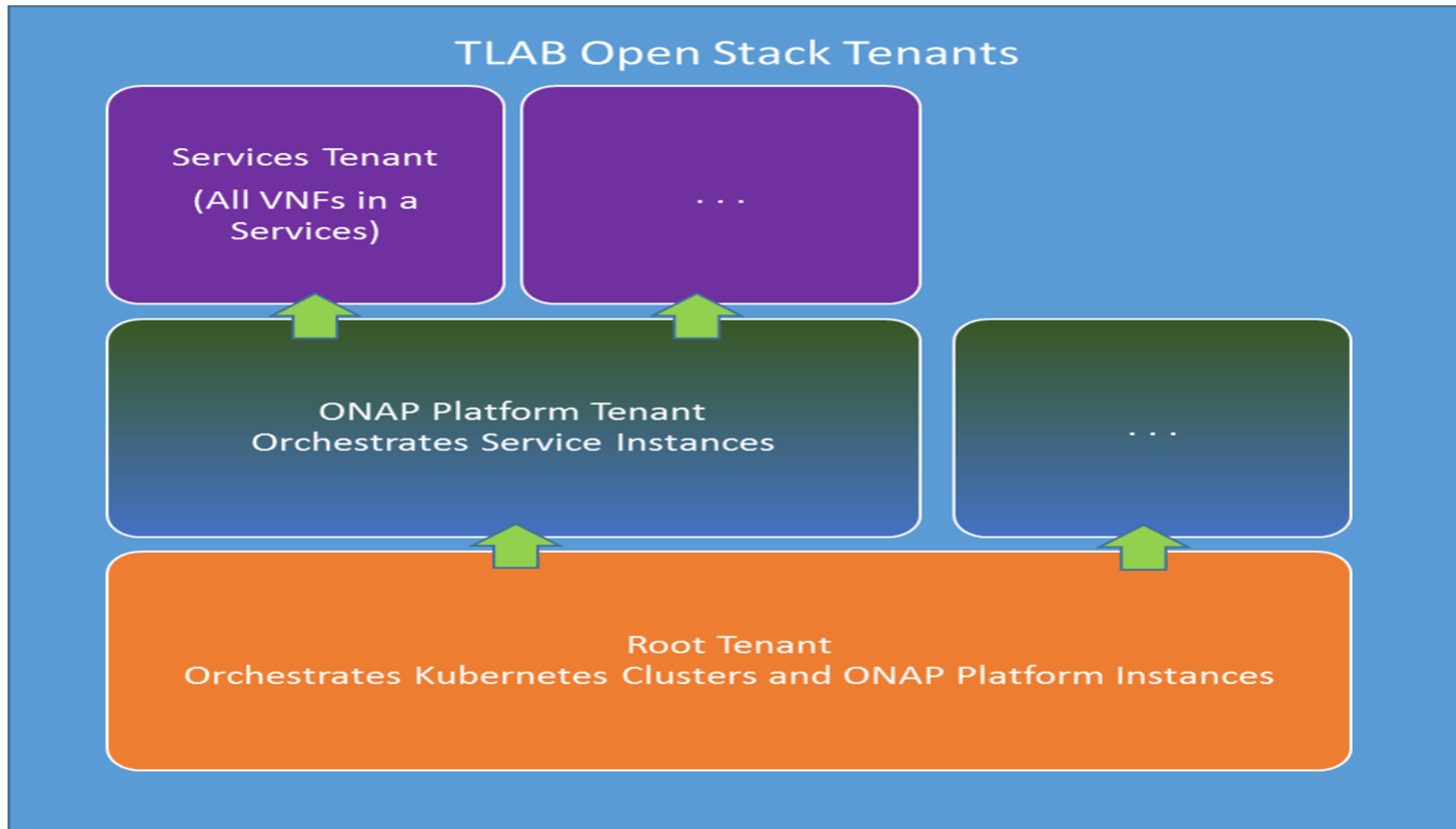


ONAP
OPEN NETWORK AUTOMATION PLATFORM

TLAB

Scott Blandford

TLAB Openstack Tenants



How to Request Access to TLAB?

- For VPN access to TLAB, create a sub-task issue on this story: <https://jira.onap.org/browse/OPENLABS-128>
- and include your...
 - Name:
 - Email:
 - Company:
 - Linux Foundation ID:
 - Reason for access (eg. OPENLAB Project TLAB Component JIRA Issue(s)):



Orange

Eric Debeau

Orange Openlab

- Who can access?
 - Any ONAP contributor (test, integration, VNF onboarding,..) can ask for an access
 - Orange can revoke the access if resources are used in an inappropriate way

- <https://wiki.onap.org/display/DW/Orange+OpenLab>

How to Request Access to OpenLab?

- Send a mail to onap-openlab@list.orange.com with:
 - Name
 - Company/university
 - Nature of the work you want to do on ONAP
 - Public ssh key
 - Start and end date
- Then ssh [xxx@onap.pod4.opnfv.fr](ssh:xxx@onap.pod4.opnfv.fr) in order to access it (you can use socks proxy this way)

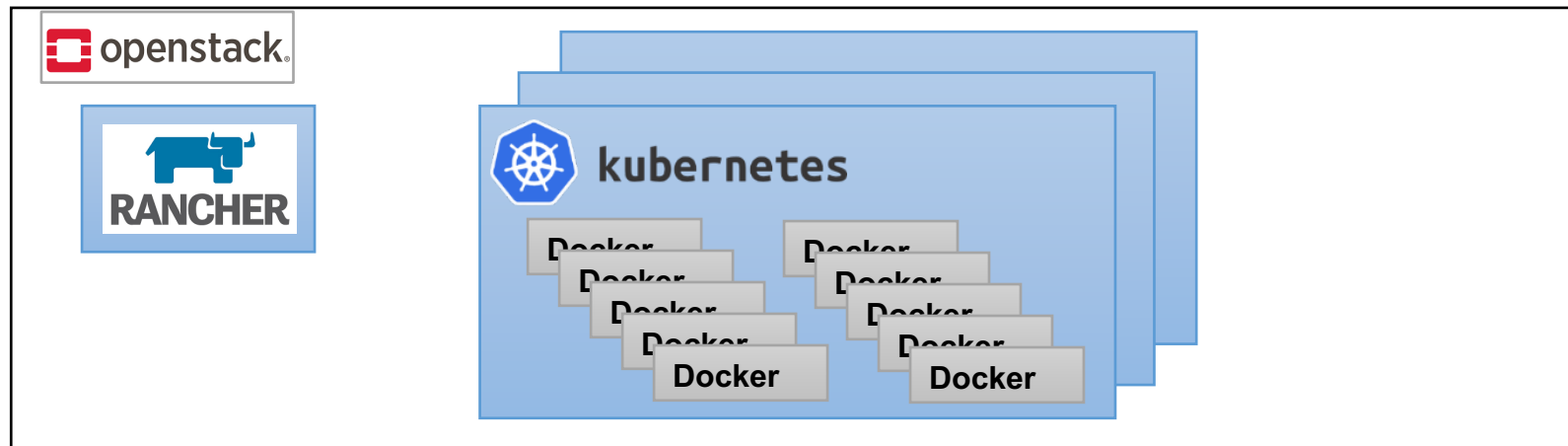


ONAP Integration Tools

Gary Wu <gary.i.wu@huawei.com>
Sr. Staff Engineer, SDN Orchestration, Huawei US R&D

Automated OOM Deployment on OpenStack

- HEAT template for OOM deployment
- Installs 1 Rancher VM and 11 k8s VMs (16 GB RAM each)
- Spins up ONAP using OOM in the k8s VMs



How to Use OOM HEAT Template, p1

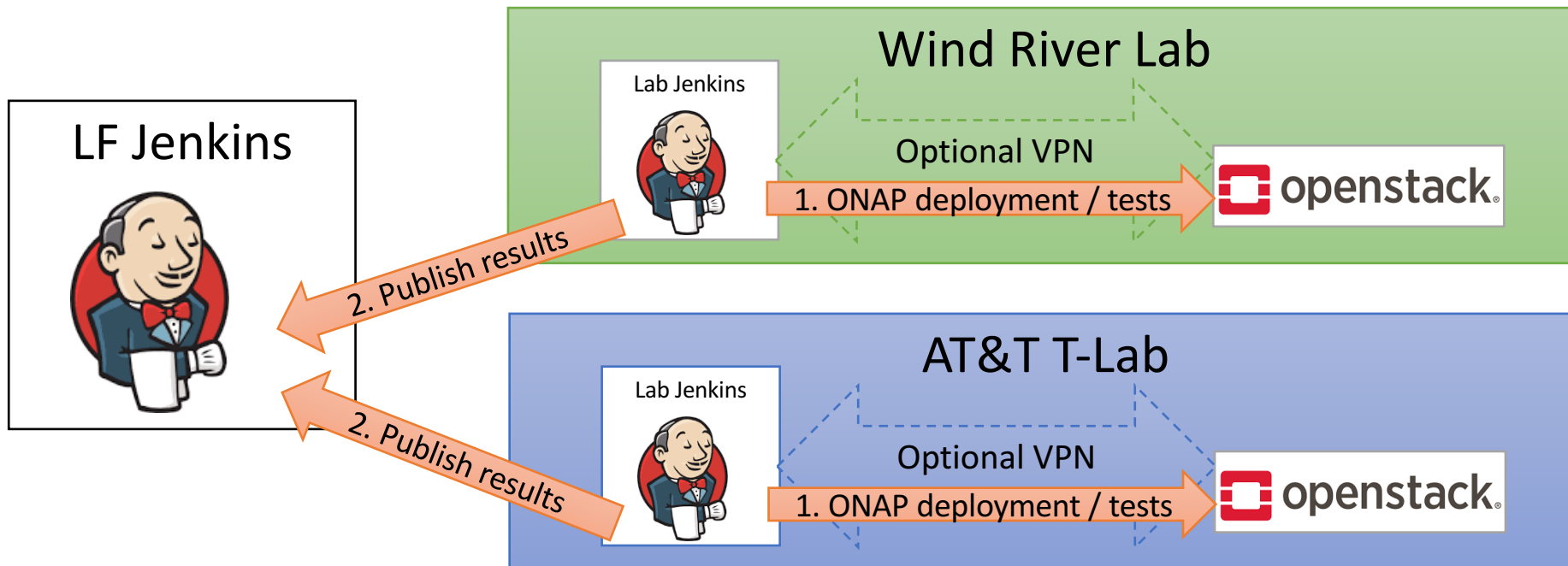
- [integration/deployment/heat/onap-oom](#)
- Configure your .env file
 - Working samples can be found in env/windriver subdirectory
- Built-in support for local apt and docker caches
 - All tenants Wind River can share the same apt/docker cache
- In-line configuration of helm integration-override.yaml
 - Enable/disable ONAP components, replicaCounts, etc.
 - Resulting yaml placed in Rancher VM /root/ directory

How to Use OOM HEAT Template, p2

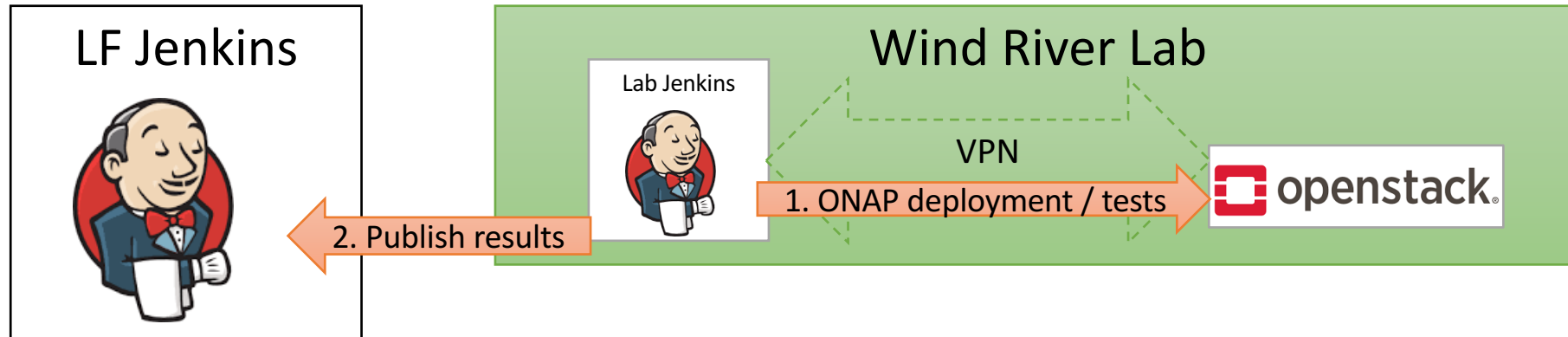
- Deploy via heat: “openstack stack create”
 - Requires the openrc file for your environment
- scripts/deploy.sh
 - deploys and runs health checks
 - NOTE: will delete everything in the tenant first

External Lab Jenkins Setup

- Lab Jenkins servers as Intermediary for each Lab
 - Internal or via VPN
- Jobs scheduled and run by Lab Jenkins
- Results published back to LF Jenkins for archival / analysis



CI Setup for Wind River Lab



- 1: Automated Deployment
 - Daily deployments to “Integration-Jenkins” and “Integration-Stable”
 - Wind River Lab Jenkins: <http://12.234.32.117/jenkins/>
- 2: Publish test results to LF Jenkins
 - <https://jenkins.onap.org/view/External%20Labs/>

External Lab Jenkins Screen Shot

The screenshot shows the Jenkins dashboard interface. At the top, there's a navigation bar with the Jenkins logo, a search bar, and a 'log out' button. Below the navigation bar, there's a sidebar on the left with various menu items like 'New Item', 'People', 'Build History', etc. The main content area displays a table of build jobs. The table has columns for status (S), weather icon (W), Name, Last Success, Last Failure, Last Duration, and Robot Results. The jobs listed include various tests and deployments for 'nexus3-docker-image-check', 'tlab-beijing-oom-deploy', 'tlab-casablanca-heat-deploy', etc. The 'Robot Results' column shows the number of tests passed out of the total number of tests.

S	W	Name ↓	Last Success	Last Failure	Last Duration	Robot Results
		nexus3-docker-image-check	16 min - #875	8 days 11 hr - #671	2 min 15 sec	
		tlab-beijing-oom-deploy	3 days 17 hr - #399	3 days 23 hr - #397	48 min	
		tlab-casablanca-heat-deploy	1 day 0 hr - #1	N/A	44 min	40 / 40 passed
		tlab-casablanca-heat-healthcheck	14 min - #94	N/A	25 sec	40 / 40 passed
		tlab-casablanca-heat-instantiate	23 hr - #1	N/A	47 min	3 / 3 passed
		tlab-casablanca-heat-stability72hr	13 hr - #11	19 min - #24	21 min	1 / 3 passed
		tlab-casablanca-oom-deploy	12 hr - #6	N/A	47 min	43 / 43 passed
		tlab-casablanca-oom-healthcheck	1 min 45 sec - #68	N/A	32 sec	43 / 43 passed
		tlab-casablanca-oom-instantiate	N/A	12 hr - #3	1 hr 29 min	
		tlab-casablanca-oom-stability72hr	2 hr 19 min - #16	1 hr 19 min - #17	45 min	
		windriver-beijing-heat-deploy	12 hr - #436	11 days - #403	2 hr 11 min	39 / 40 passed
		windriver-beijing-heat-healthcheck	4 min 45 sec - #14019	3 days 21 hr - #13689	17 sec	39 / 40 passed
		windriver-beijing-heat-instantiate	10 hr - #58	2 days 2 hr - #55	39 min	3 / 3 passed
		windriver-beijing-heat-stability72hr	2 hr 19 min - #67	1 hr 19 min - #68	24 min	2 / 3 passed
		windriver-beijing-oom-deploy	3 days 17 hr - #479	11 days - #453	1 hr 3 min	43 / 43 passed
		windriver-beijing-oom-healthcheck	13 min - #9192	3 days 5 hr - #8883	20 sec	43 / 43 passed
		windriver-beijing-oom-instantiate	3 days 16 hr - #49	3 days 19 hr - #48	2 hr 7 min	3 / 3 passed

External Lab Jenkins Job Types

- Various job types to assist integration testing, prepare testing environments, etc.
- deploy
 - One-click complete redeployment of OOM or HEAT
- healthcheck
 - Runs health checks against the instance every 15 minutes
- instantiate
 - Automatically runs after deployment completion
 - Runs `demo.sh init`, `ete.sh healthdist`, `ete.sh distribute`, `ete.sh instantiate`
- stability72hr
 - Runs `ete.sh stability72hr` hourly

Casablanca Plans (Tentative)

- Support Offline Deployment
 - Allow ONAP deployment without internet access
- Improved Docker image build process
 - Incremental builds, taking advantage of cached FS layers, etc.
- More automation in the CI/CD flow
 - Improved verification jobs to prevent bad merges
 - “Known Good” vs “Experimental” docker manifests
 - More comprehensive automated test flows (vFWCL, vCPE, etc.)
- Consider use of Zuul for better CI validation and Spinnaker for CD
- Propose a dashboard on last deployments success/failures like OPNFV

Open Questions

- Do we continue to have OOM **and** HEAT based deployment or OOM only to simplify the effort?
- If OOM only, how to move CSIT tests that are “HEAT like” to a “OOM like” model?
- Do we consider upgrade automated tests to test Beijing → Casablanca upgrade?

- We need at least one community lab from Asian region:
 - Hardware requires:
 - Minimum 2 TB RAM, 20 TB Disk (for 10 ONAP instances)
(1 ONAP instance for Beijing: 200 GB RAM, 2000 GB Disk)
(Current Wind River: Vcpu: 550 / RAM: 5TB / Local Disk: 53TB)
 - Access: Remote
 - Support: 24/5
 - More specific requirements:
<https://wiki.onap.org/pages/viewpage.action?pageId=5735793>
- Email: onap-discuss, with [openlab][integration] tags.



ONAP

OPEN NETWORK AUTOMATION PLATFORM

谢谢

ONAP Infrastructure

- Testing environment: currently we have **two labs ready** for ONAP community to use for end to end integration testing and pairing testing
 - Intel / Wind River Lab
 - TLAB
- Tools (under investigating)
 - Performance / Scalability: JMeter / Locust
 - Profiling: JProfile (for Java code only)
 - Resilience: Chaos Monkey
 - Security: Sonarqube, Bandit, Nexus Auditor, Nmap, Burp suite
 - Stability: we plan to write some python scripts
- Simulators for controller and VNFM

