ONAP
OPEN NETWORK AUTOMATION PLATFORM

Kubernetes vs. Dockers Swarm

Supporting ONAP
On Multi-Cloud Multi-Stack Environment

ONAP Operations Manager(OOM)

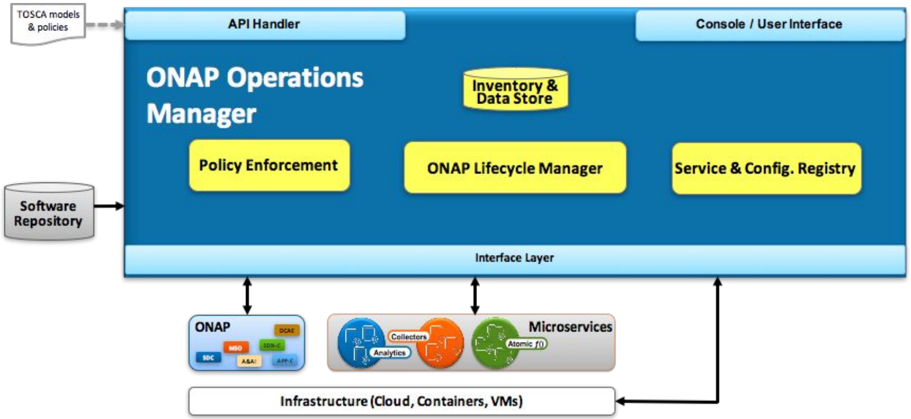
Arthur Berezin, Cloudify
Sep 28 , 2017

Agenda

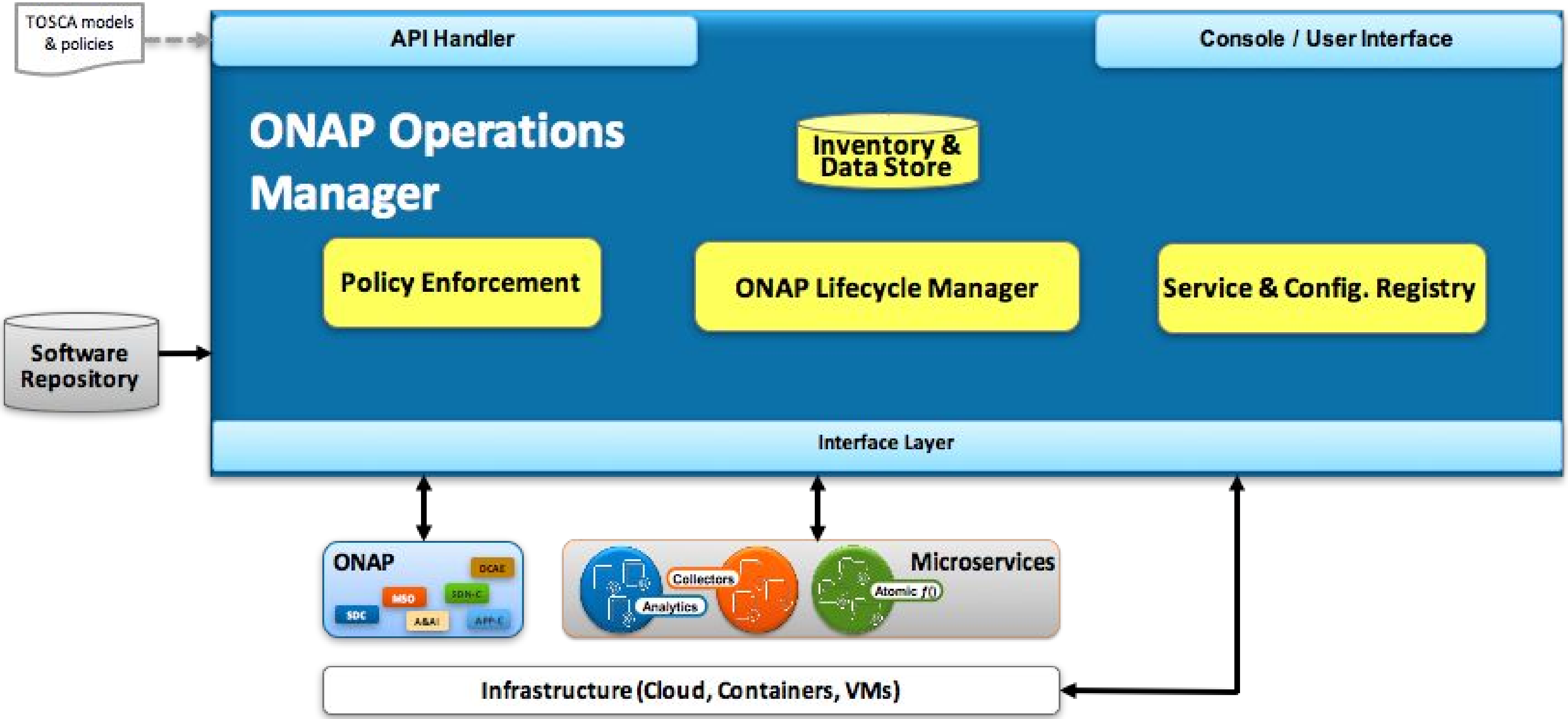
1. **Setting the context:**
   ONAP Operations Manager (OOM)
2. **Kubernetes vs. Docker Swarm**
3. **How TOSCA Can Help?**
   a. Integration
   b. Future Proofing
   c. Reusing Technologies
4. **ONAP OOM R2 Proposal**

Context: ONAP Operations Manager

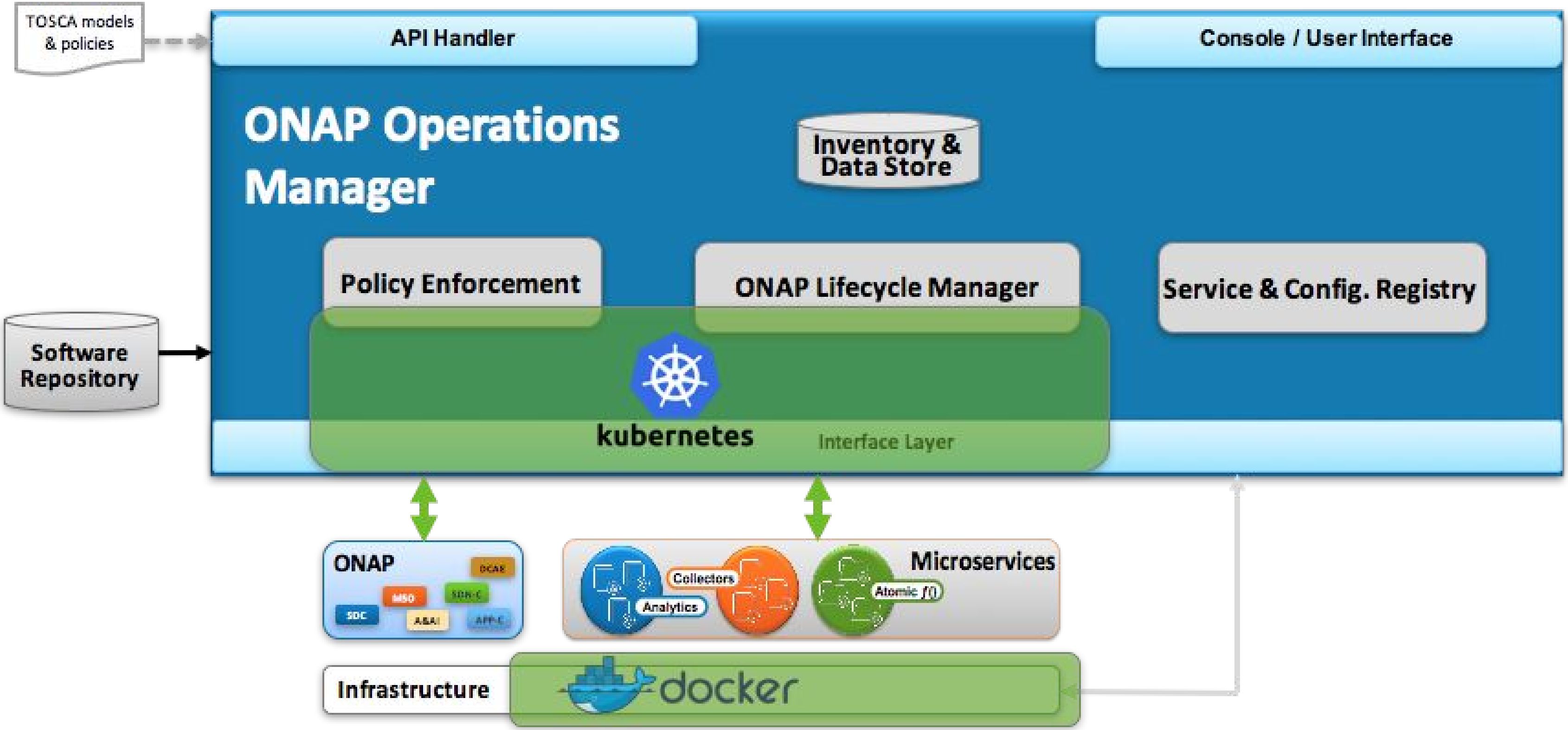
- Flexible Platform Deployment
- State Management of ONAP platform components
- Platform Operations Orchestration / Control Loop Actions



Context: ONAP Operations Manager

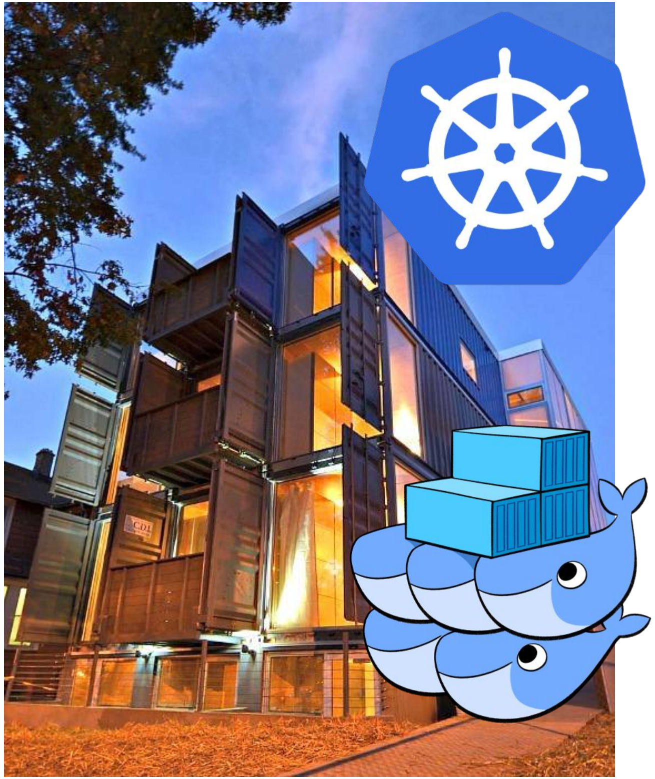


Context: ONAP Operations Manager - R1



Container Orchestration

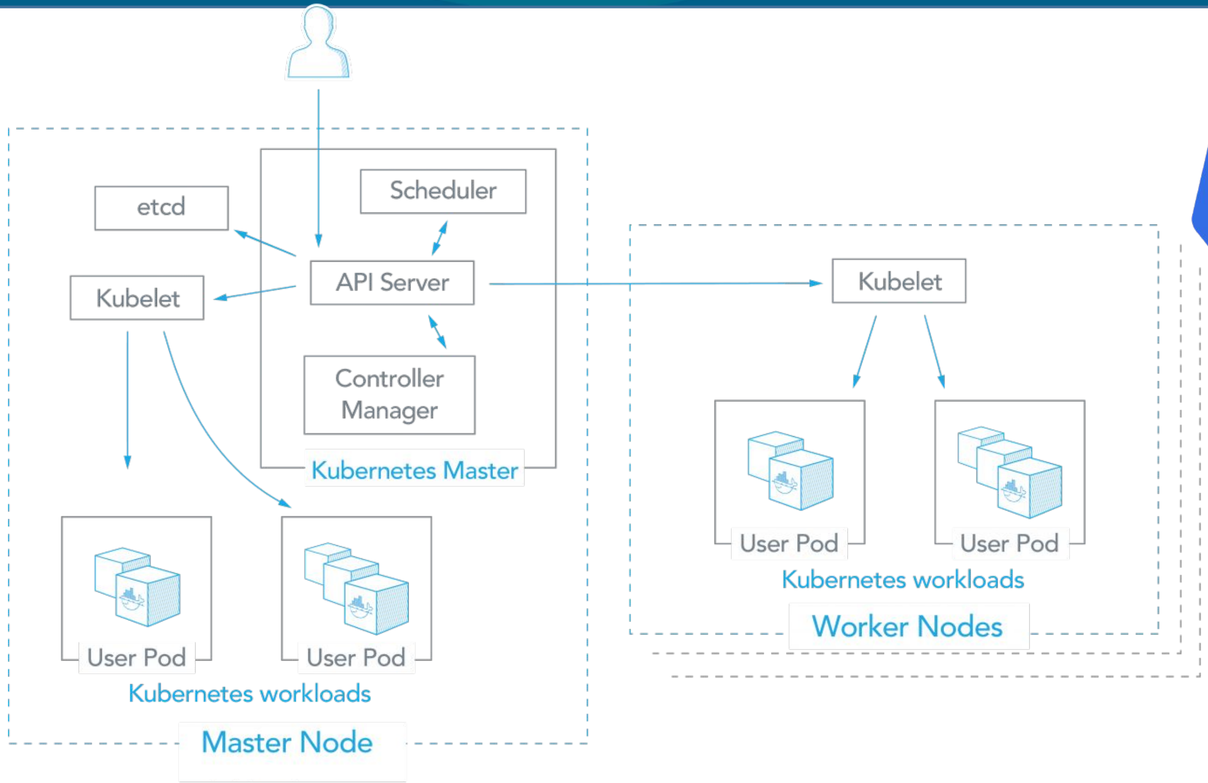
- Multiple Hosts
- Placement Control
- Networking
- Affinity/Anti-Affinity
- High Availability
- Scaling
- Load Balancing
- Rolling Upgrades



Kuberneres

- Open sourced by Google
- Master/worker
- Pod unit of deployment/scale
- Replication Controller (autoheal/scale)
- Service support with LB external IP
- Overlay networks (IP per pod) + DNS
- No supported master HA
- Container agnostic
- Placement, affinity + anti-affinity
- YAML deployment model

Kuberneres

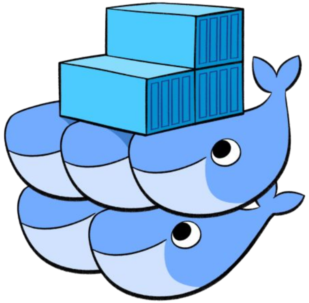


API Server: management hub for Kubernetes
Scheduler: places a workload on the appropriate Node
Controller Manager: scales workloads up/down
etcd: stores configuration data which can be accessed by API Server

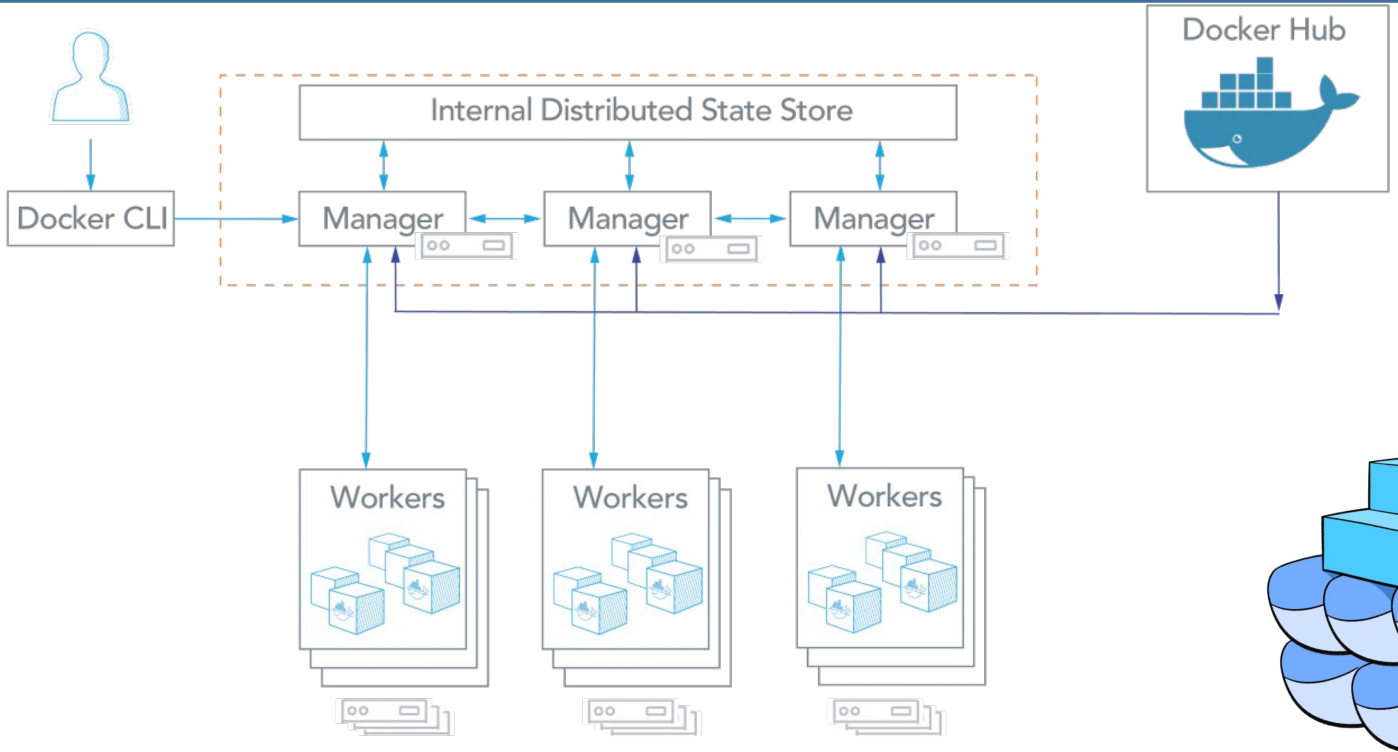
Kubelet: Receives pod specifications from API Server, updates Nodes
Master Node: places workloads on Nodes
Worker Nodes: receives requests from Master Nodes and dispatches them
User Pod: a group of containers with shared resources

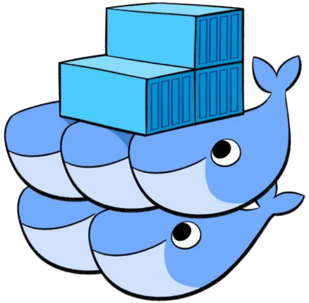
Docker Swarm 1.12+

- Integrated into Docker engine
- Manager/worker
- Auto heal, manual scale
- Service support with LB fixed IP
- Overlay networks & DNS
- Highly available manager
- Network security (TLS) with CA
- Node placement + Affinity/anti-affinity
- Deployment modeling via Compose



Docker Swarm 1.12+





Manager: a node that dispatches tasks
Worker: a node that executes tasks provided by a Manager
Internal Distributed Store: used to maintain cluster state

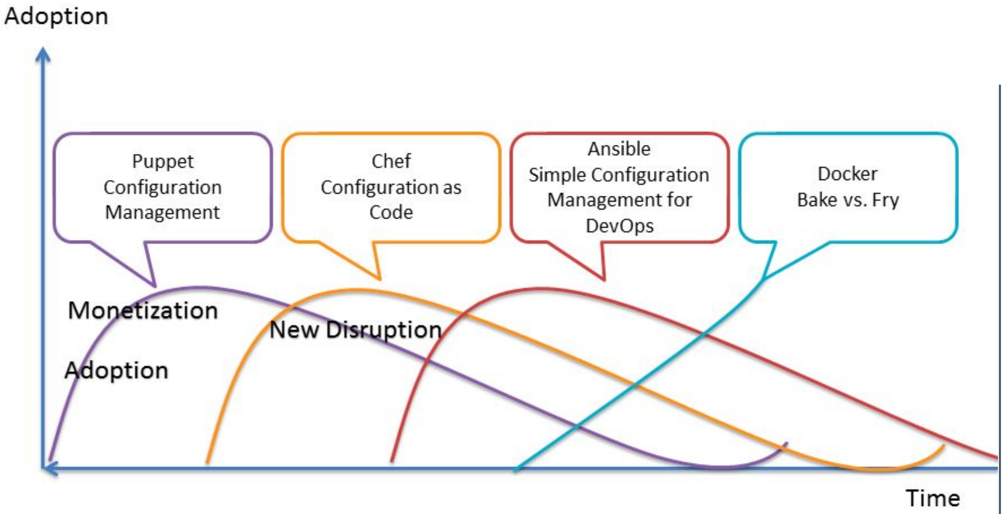
Docker CLI: User interacts with the swarm using Docker CLI, for example "docker service"
Docker Hub: contains repositories for downloading and sharing container images

How Do They Compare?

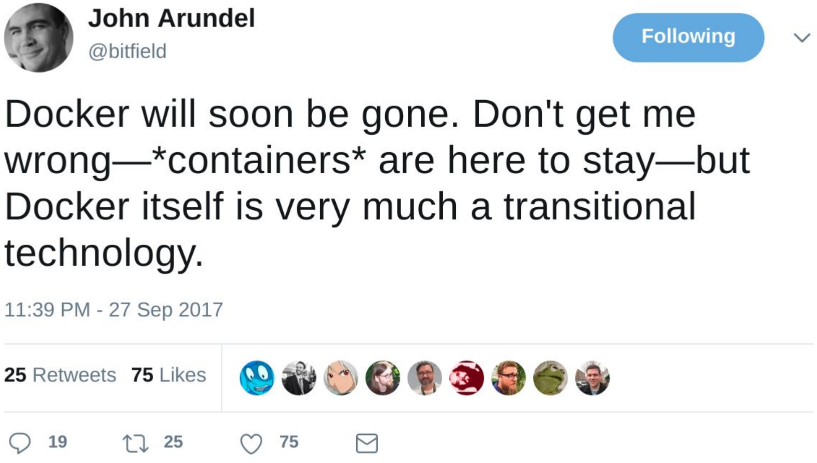
- Swarm Setup Super Simple **a+**
- Lack of Pod in Swarm **a –**
- Lack of HA in Kubernetes **a –**
- Swarm tie to Docker may be **a –**
- Swarm lack of autoscale may be **a –**
- Swarm built in security **a +**
- Swarm playing catchup (service/replicas)
  But also has some cool features

Waves Of Changes Are Getting Quicker

Disruption is Everywhere!



...I was using this slide in 2016...



John Arundel
@bitfield

Following

Docker will soon be gone. Don't get me wrong—*containers* are here to stay—but Docker itself is very much a transitional technology.

11:39 PM - 27 Sep 2017

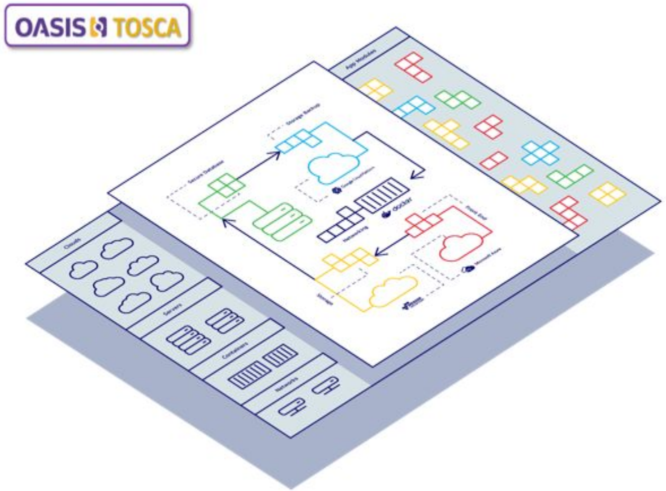
25 Retweets **75** Likes

19 25 75

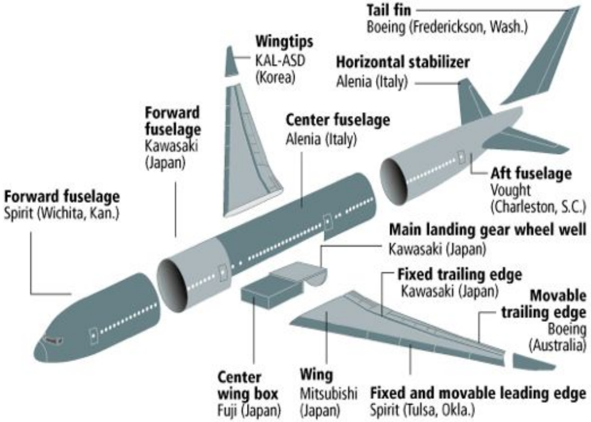
… Meanwhile in 2017 (Only 12 months later)
ONAP VVP(ICE) is already using Rkt Containers

Standard Modeling == Automation at Scale

Use what's already working, but be ready for changes



Example from the Manufacturing Industry



The Boeing 787 could not be produced
without standard modeling

ONAP Already Uses TOSCA for VNFs and Network Services Orchestration

ONAP

OPEN NETWORK AUTOMATION PLATFORM

TOSCA Basic Concepts

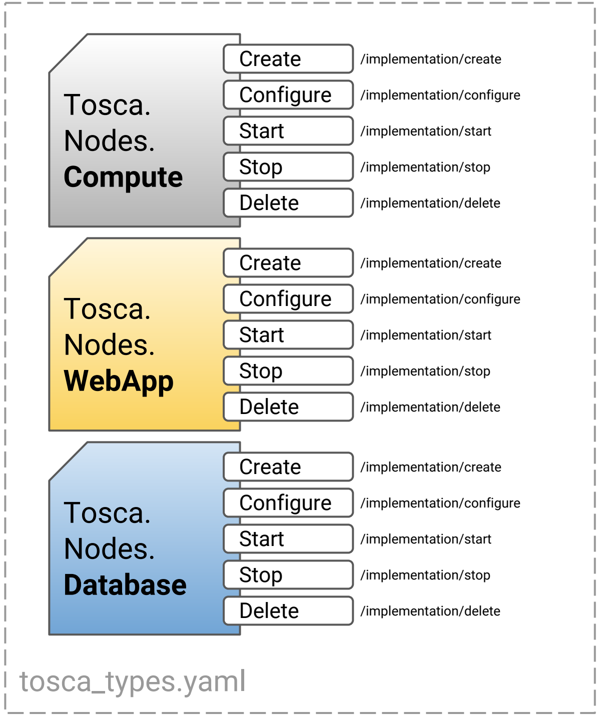
Sep 28 , 2017

Model Driven Service Orchestration - Types

Node Types

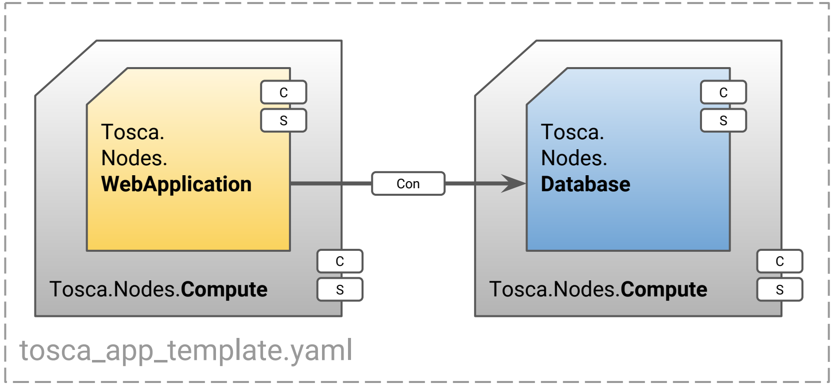
- Properties
- Attributes
- Lifecycle Interfaces

- Normative node types (Compute, Network, etc’)
- Relationship types.
- Lifecycle operations and implementations (create, configure, start and others)
- ARIA supports custom types.



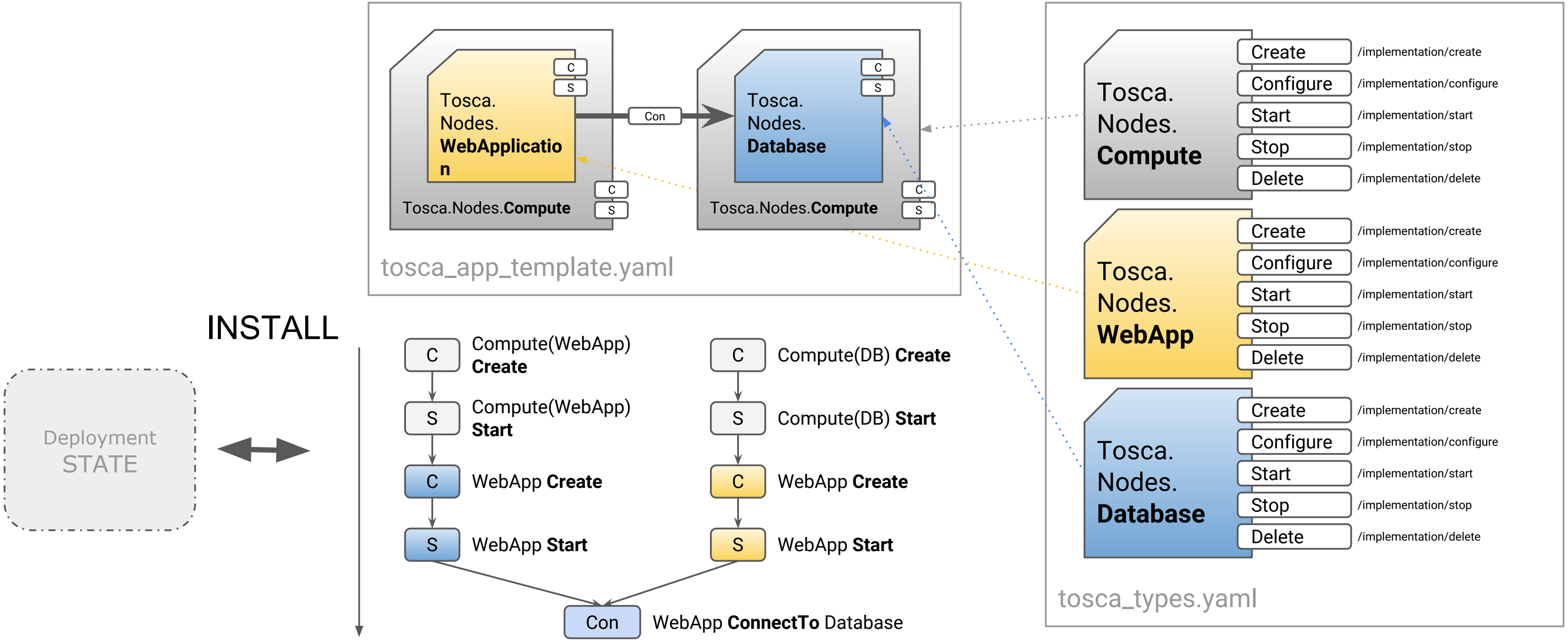
TOSCA Application Topology

Containment
Connection



- TOSCA Template describes the topology of the application.
- Topology templates use TOSCA node types describing the nodes and their relationships using normative and custom types.
- Templates can also define implementations for lifecycle operations.

Declarative Model-Driven Orchestration



Tosca.
Nodes.
**WebApplicatio
n**

C
S

Tosca.Nodes.**Compute**

C
S

Con

Tosca.
Nodes.
Database

C
S

Tosca.Nodes.**Compute**

C
S

tosca_app_template.yaml

Tosca.
Nodes.
Compute

Create	/implementation/create
Configure	/implementation/configure
Start	/implementation/start
Stop	/implementation/stop
Delete	/implementation/delete

Tosca.
Nodes.
WebApp

Create	/implementation/create
Configure	/implementation/configure
Start	/implementation/start
Stop	/implementation/stop
Delete	/implementation/delete

Tosca.
Nodes.
Database

Create	/implementation/create
Configure	/implementation/configure
Start	/implementation/start
Stop	/implementation/stop
Delete	/implementation/delete

tosca_types.yaml

INSTALL

Deployment
STATE

C Compute(WebApp) **Create**

S Compute(WebApp) **Start**

C WebApp **Create**

S WebApp **Start**

C Compute(DB) **Create**

S Compute(DB) **Start**

C WebApp **Create**

S WebApp **Start**

Con WebApp **ConnectTo** Database

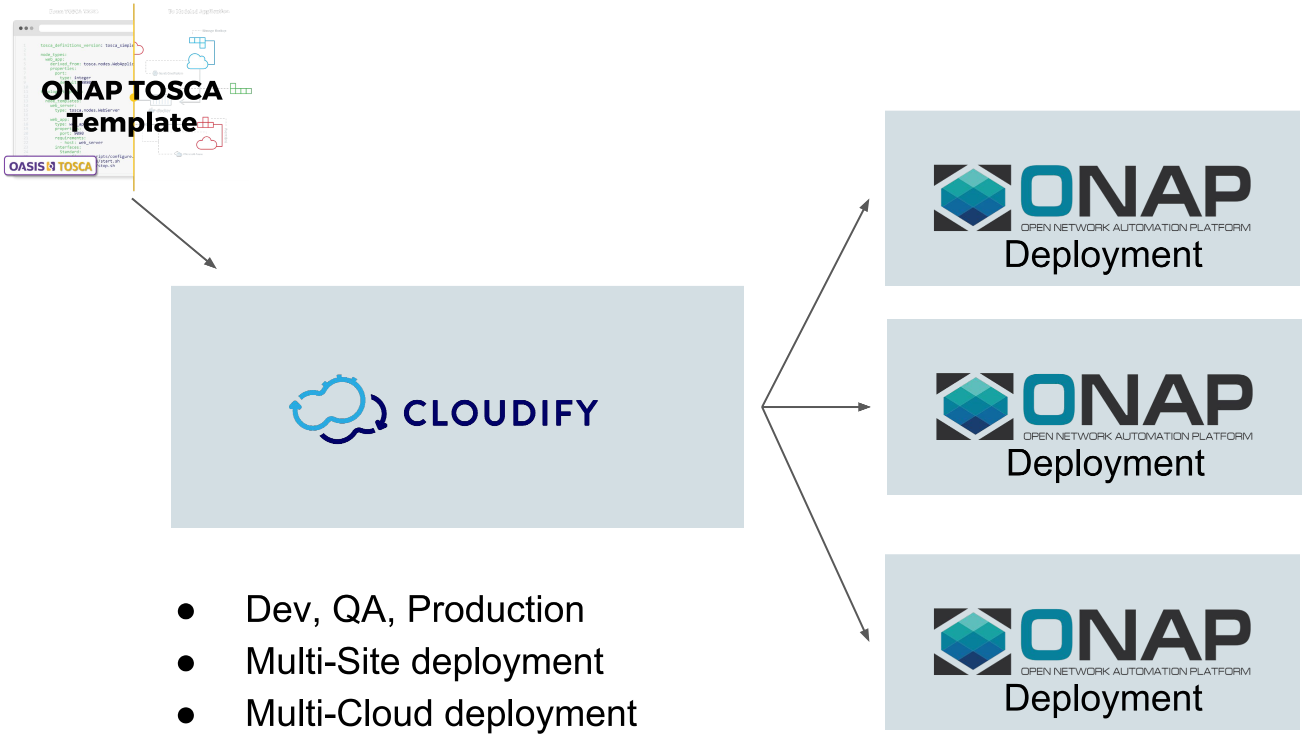
ONAP

OPEN NETWORK AUTOMATION PLATFORM

ONAP OOM TOSCA Template

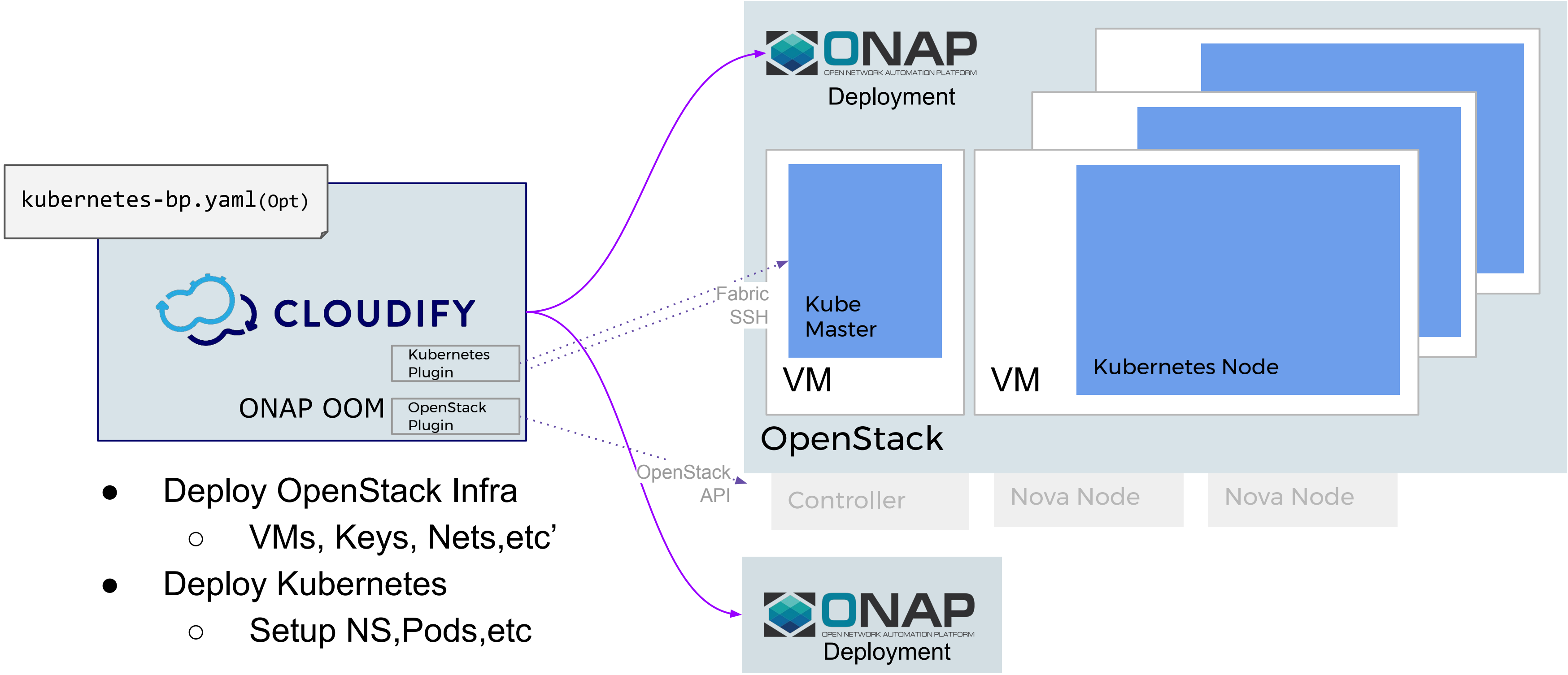
Sep 28 , 2017

Allowing Multiple ONAP Deployments


ONAP TOSCA Template
OASIS TOSCA
CLOUDIFY
ONAP OPEN NETWORK AUTOMATION PLATFORM
Deployment
ONAP OPEN NETWORK AUTOMATION PLATFORM
Deployment
ONAP OPEN NETWORK AUTOMATION PLATFORM
Deployment

- Dev, QA, Production
- Multi-Site deployment
- Multi-Cloud deployment

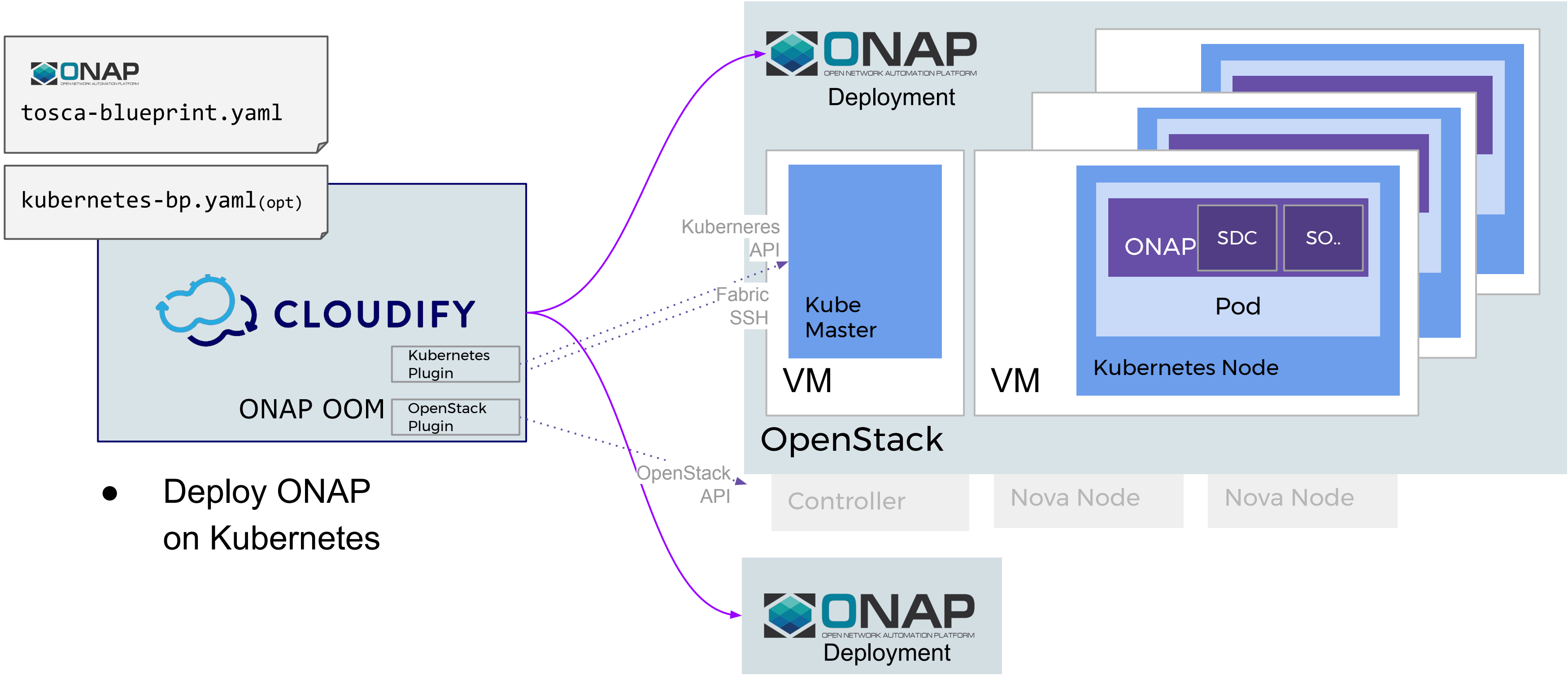
Deploy Infrastructure - OpenStack, K8ns, etc’



- Deploy OpenStack Infra
  - VMs, Keys, Nets,etc’
- Deploy Kubernetes
  - Setup NS,Pods,etc

https://gerrit.onap.org/r/gitweb?p=oom.git;a=blob;f=onap-blueprint.yaml

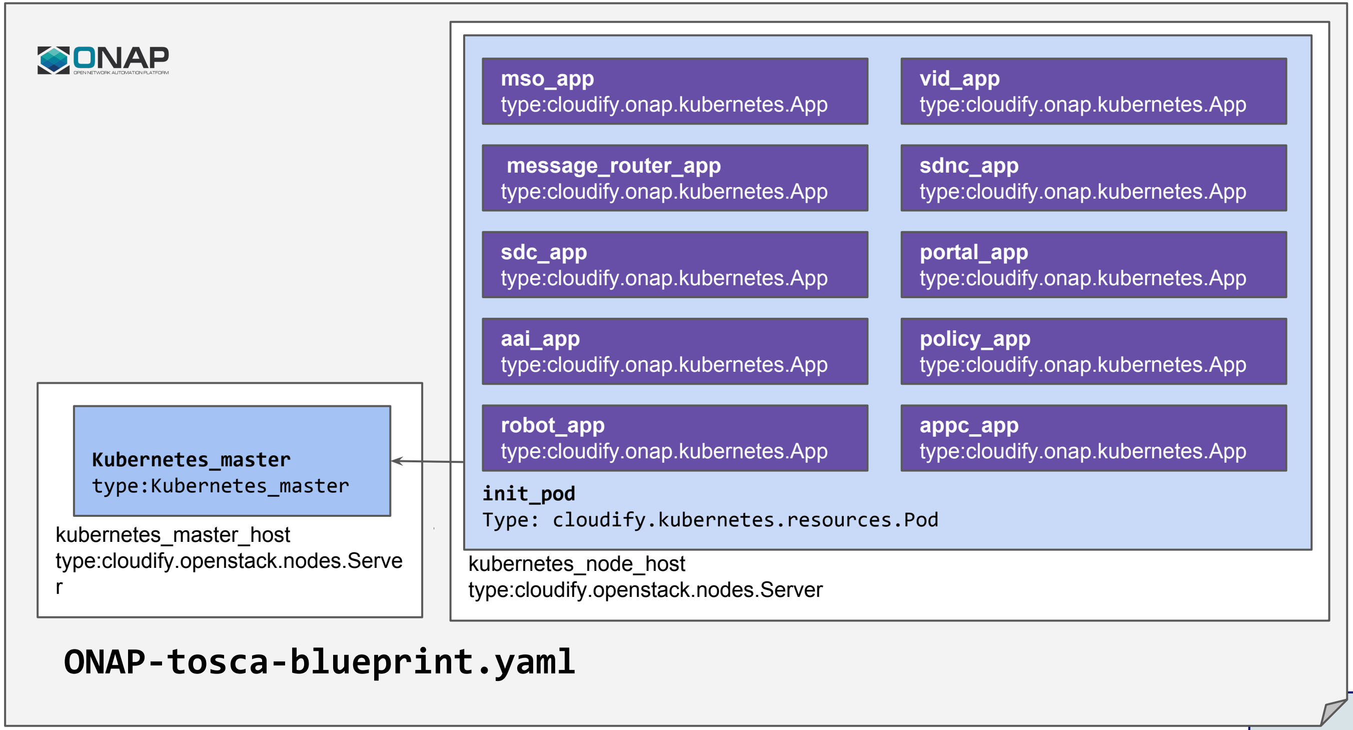
Deploy ONAP



- Deploy ONAP on Kubernetes

https://gerrit.onap.org/r/gitweb?p=oom.git;a=blob;f=onap-blueprint.yaml

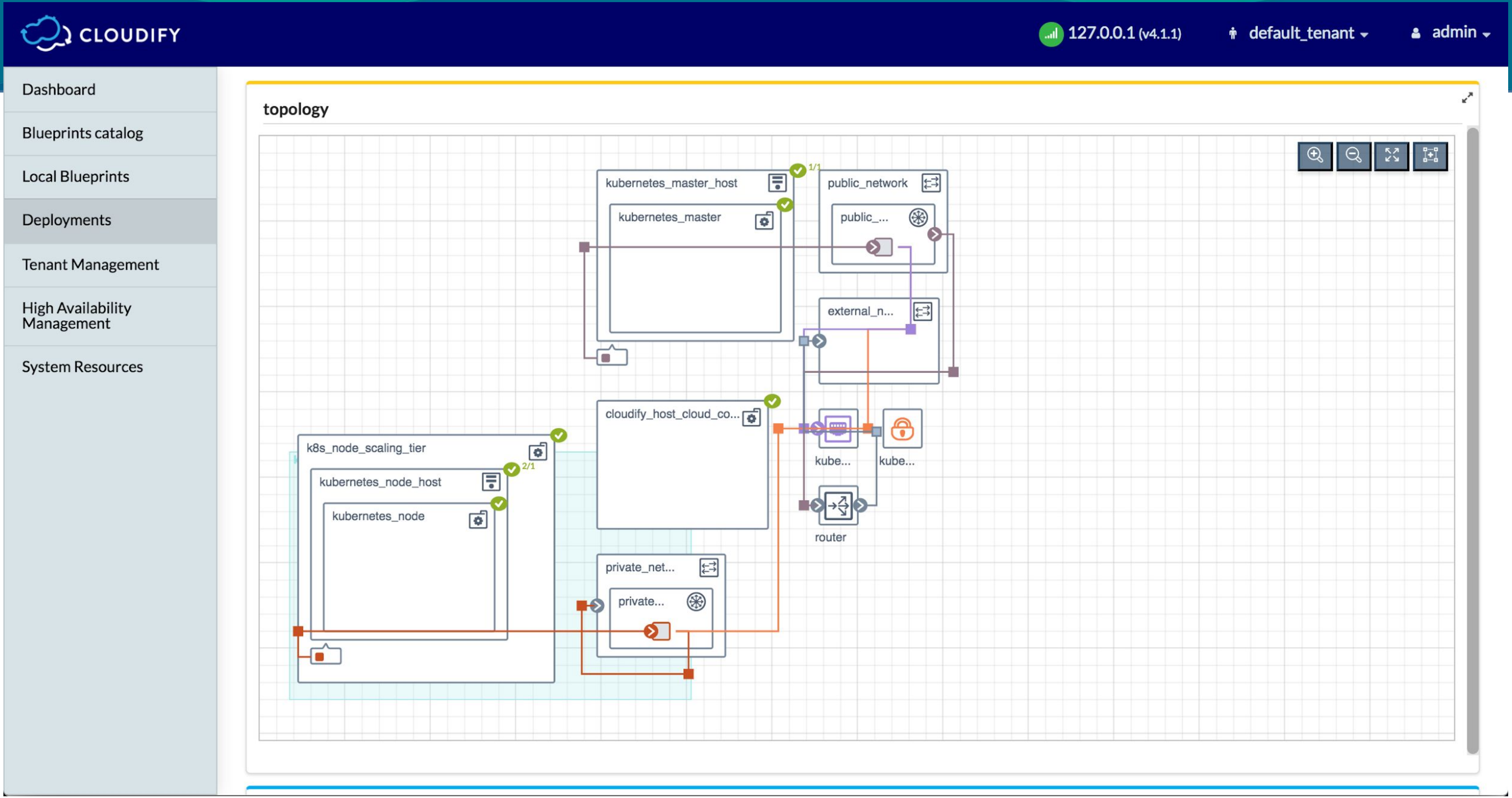
ONAP TOSCA Template


ONAP
OPEN NETWORK AUTOMATION PLATFORM
mso_app
type:cloudify.onap.kubernetes.App
vid_app
type:cloudify.onap.kubernetes.App
message_router_app
type:cloudify.onap.kubernetes.App
sdnc_app
type:cloudify.onap.kubernetes.App
sdc_app
type:cloudify.onap.kubernetes.App
portal_app
type:cloudify.onap.kubernetes.App
aai_app
type:cloudify.onap.kubernetes.App
policy_app
type:cloudify.onap.kubernetes.App
robot_app
type:cloudify.onap.kubernetes.App
appc_app
type:cloudify.onap.kubernetes.App
init_pod
Type: cloudify.kubernetes.resources.Pod
kubernetes_node_host
type:cloudify.openstack.nodes.Server
Kubernetes_master
type:Kubernetes_master
kubernetes_master_host
type:cloudify.openstack.nodes.Server
r
ONAP-tosca-blueprint.yaml

https://gerrit.onap.org/r/gitweb?p=oom.git;a=blob;f=onap-blueprint.yaml

CLOUDIFY

(ONAP OOM)


CLOUDIFY
127.0.0.1 (v4.1.1)
default_tenant
admin
Dashboard
Blueprints catalog
Local Blueprints
Deployments
Tenant Management
High Availability Management
System Resources
topology
kubernetes_master_host
1/1
kubernetes_master
public_network
public_...
external_n...
cloudify_host_cloud_co...
k8s_node_scaling_tier
kubernetes_node_host
2/1
kubernetes_node
kube...
kube...
router
private_net...
private...

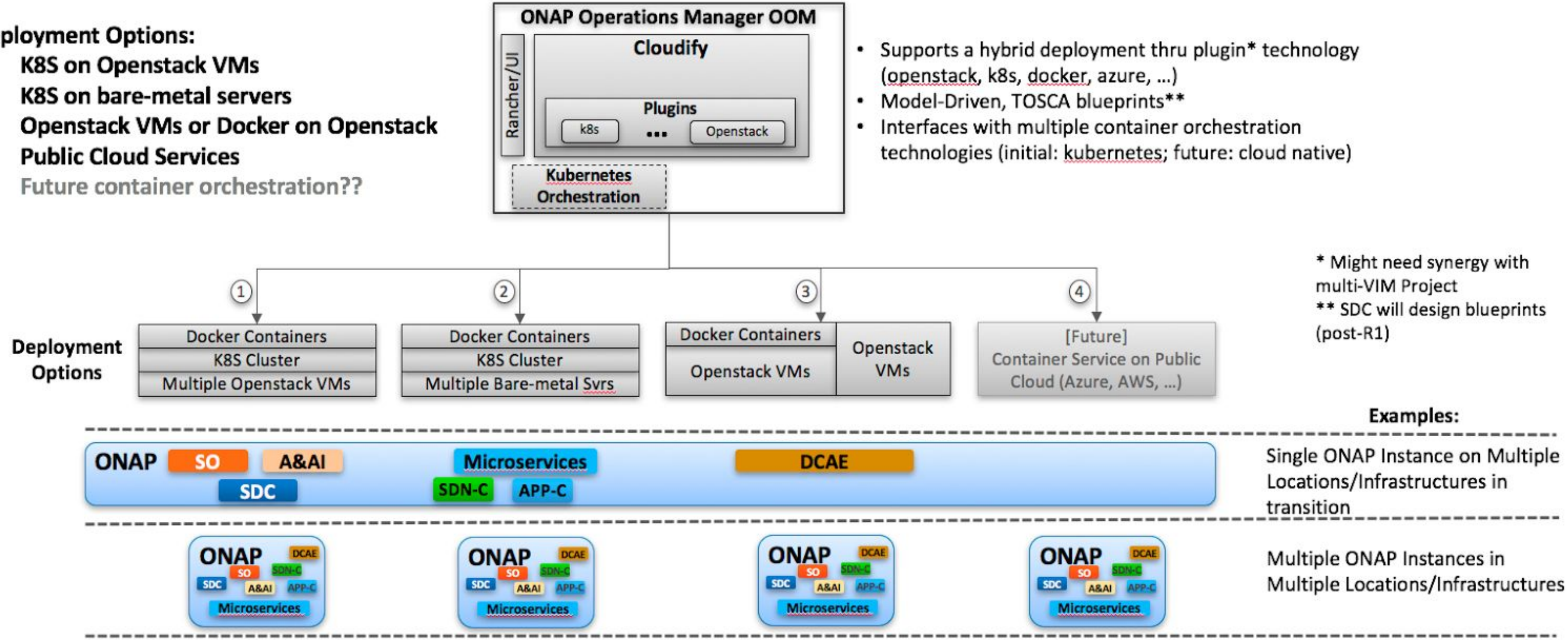
Benefits of Using TOSCA in OOM

- Mixing and Matching Technologies
- Manage the different layers of the stack
  - Bare-Metal, OpenStack, Kubernetes,
- Use or deploy additional software stacks or infrastructure as needed
  - {For Example use external HAMariaDB with Galera: https://github.com/cloudify-examples/mariadb-blueprint }
- Allow the operator or distro choose the stack

Summery

Deployment Options:

1. **K8S on Openstack VMs**
2. **K8S on bare-metal servers**
3. **Openstack VMs or Docker on Openstack**
4. **Public Cloud Services**
5. **Future container orchestration??**



- Supports a hybrid deployment thru plugin* technology (openstack, k8s, docker, azure, ...)
- Model-Driven, TOSCA blueprints**
- Interfaces with multiple container orchestration technologies (initial: kubernetes; future: cloud native)

* Might need synergy with multi-VIM Project

** SDC will design blueprints (post-R1)

Examples:

Single ONAP Instance on Multiple Locations/Infrastructures in transition

Multiple ONAP Instances in Multiple Locations/Infrastructures

Summery

ONAP TOSCA Template is offered as part of the ONAP OOM project

- **Deploy K8s** cluster on OpenStack for ONAP
- **Use Existing K8s** cluster which allows the flexibility to connect
- **Manage k8s cluster** from within ONAP framework
- **Healing and Scaling of K8s nodes/Pods is included**
- **ONAP Services Deployment** (SO,SDC,DCAE,others) through TOSCA
- **Dynamic Deployments** by leveraging the TOSCA relationships, inputs etc'
- **Single pane of glass** for the ONAP Stack, managing K8s and non K8S services such as stateful services.
- **Use same TOSCA operational model for running NFV and ONAP cluster**
- **Mature Multi Cloud support** (OpenStack, K8S, VMware, AWs, etc..)

Thanks

Arthur Berezin, Cloudify
Sep 28 , 2017