Installing OpenShift Enterprise multi-nodes config using VirtualBox for extended demos

Gabriel Bechara

Principal Solution Architect @ Red Hat

https://github.com/gbechara/osedevops

Prerequisites

- Virtual Box (tested on Version 5.0.20 r106931)
- Vagrant (tested on 1.8.1)
- Valid RHEL7 & OpenShift Container Platform Subscription
  - Trials are available on https://www.openshift.com/container-platform/trial.html
- Create your own Vagrant Box
  - Create a VirtualBox with RHEL7 “Server”
  - Add to this VirtualBox your valid Subs or use vagrant-registration plugin
  - Convert this Box into a Vagrant Base Box located on your local disk

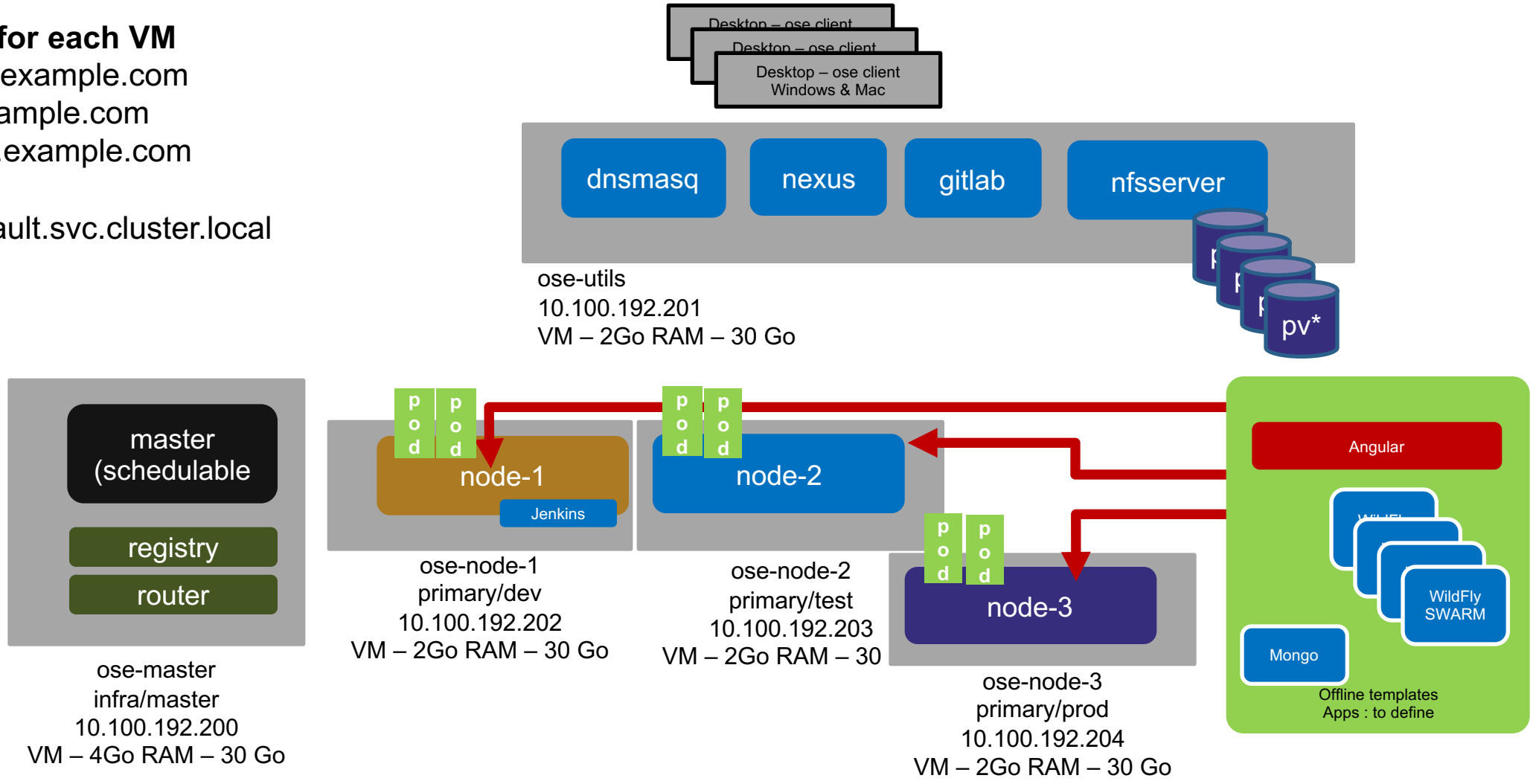
    Instruction are here https://www.vagrantup.com/docs/boxes/base.html

    name this base box rhel72-server-base.box (or change the name in the Vagrantfile)

    do not share this box : it contains your subscriptions

Architecture

DNS entry for each VM
ose-master.example.com
ose-utils.example.com
ose-node-1.example.com
Wildcards
*.router.default.svc.cluster.local



Usage (1/2)

- Get the souce code

  > git clone https://github.com/gbechara/osedevops.git

- If your Sub is not in the Vagrant box you can use the vagrant-registration plugin

  add in ~/.vagrant.d/Vagrantfile or in the Vagrantfile you got form github the following

```
Vagrant.configure('2') do |config|
   config.registration.username = '<your Red Hat username>'
   config.registration.password = '<your Red Hat password>’
   config.registration.pools = [ ’thepoolthatcontailnstheadequatesubs' ]
End
```

- Launch

  > Change the passwords in the file Vagrantfile to match the root password of your box

  This password is needed to copy the generated ssh key to all OpenShift nodes

  > cd osedevops

  > vagrant plugin install vagrant-cachier

  > vagrant plugin install vagrant-registration (if your sub is not in the vbox)

  > vagrant up

- configure your host to add a dnsserver

  on linux add in /etc/resolv.conf

  search example.com

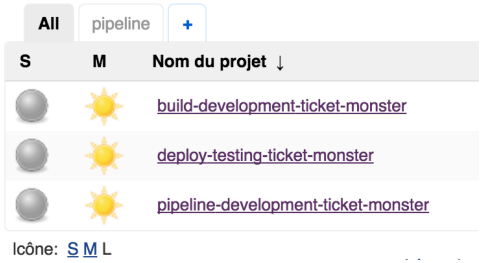
  10.100.192.201

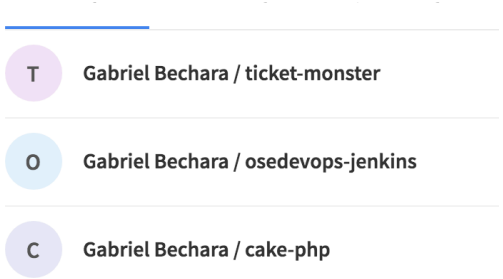
  on osx create 2 files named ‘example.com’ and ‘router.default.svc.cluster.local’ in /etc/resolver add to those files

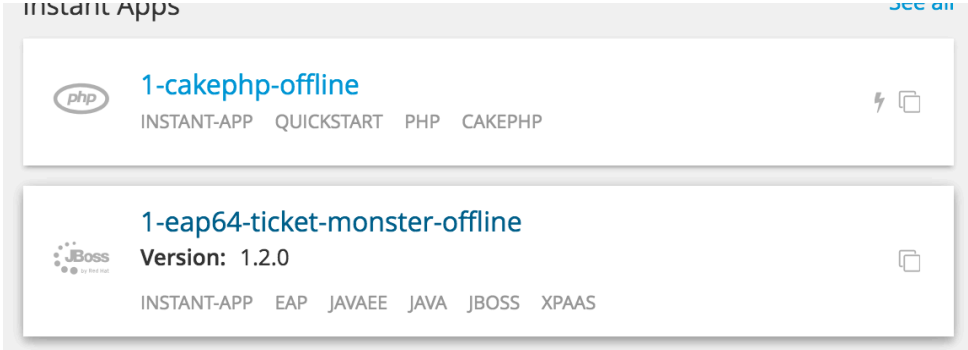
  nameserver 10.100.192.201

Usage (2/2)

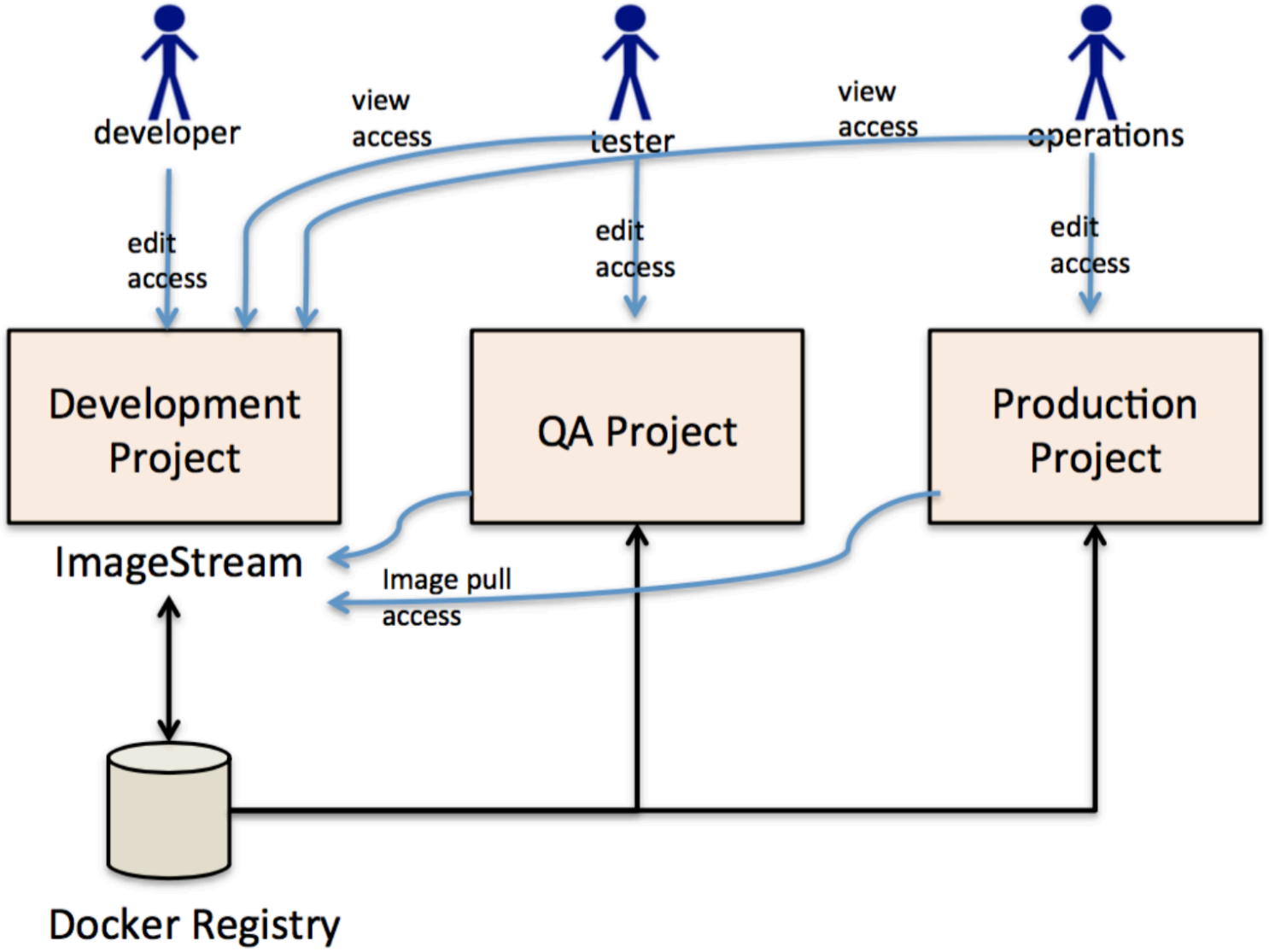
- Openshift Web Console
  - https://ose-master.example.com:8443/console/
  - User dev1/dev1 have access to the development project
  - User test1/test1 have access to development, testing, ci and production (when created) project
- Jenkins
  - https://jenkins-ci.router.default.svc.cluster.local/
  - Jenkins user is system/password
  - Use **pipeline-development-ticket-monster/**
- GitLab
  - http://gitlab.example.com/
  - User gabriel/weareawesome
  - 2 projects used by offline templates and one for jenkins







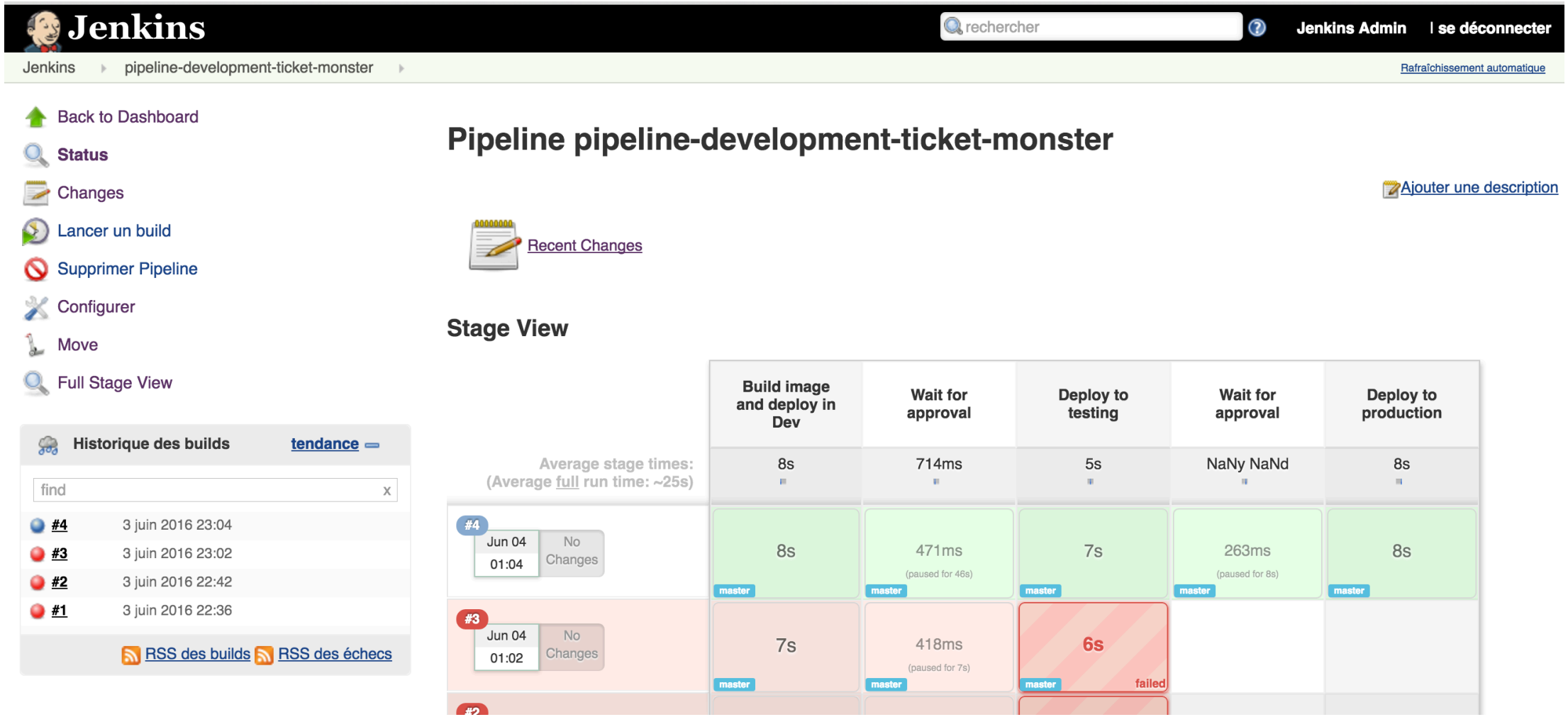
Example



Based on https://blog.openshift.com/promoting-applications-across-environments/

Jenkins Pipeline : from dev to production

→ Approvals between environments and creation of the next env if it does not exist



Notes

- Installation works only when connected to internet
  - OOTB Templates use github and maven repos
  - I may take time, around 40 minutes, depending on the roles you add
- 2 templates can then be used to do offline demos
  - 1-eap64-ticket-monster-offline
  - 1-cakephp-offline
- During the install the sample gitlab, nexus, the docker images, the jenkins plugins are all populated to work offline for the 2 templates
- 3 users (gabriel, dev1, test1), 2 projects (development, testing) are created to deploy the sample, the jenkins job “pipeline-development-ticket-monster” will create a third project

Roadmap

- ✓ Add NFS for persistent Volumes
- ✓ Add a git server
- ✓ Add a nexus
- ✓ Add templates for offline demos using the git server and the nexus
- ✓ Deploy and example to populate the docker registry, gitlab and nexus
- ✓ Showcase an application promotion across environments
- ✓ Showcase Jenkins with approval steps
- Add other samples
  - Deploying WildFly SWARM microservices on OpenShift
  - Covering the entire DevOps lifecycle
  - Other ideas ?
  - ….