Red Hat

WILDFLY AND KUBERNETES HOW THEY PLAY TOGETHER

ONDRA CHALOUPKA

http://narayana.io, @_chalda

Speaker notes

Demo is simple but shows the multiple Java EE technologies. We will touch them with concern of the journey to run them in Kubernetes.

AGENDA
WILDFLY TO KUBERNETES

- s2i builds
- Galleon provisioning tool
- WildFly Operator

Speaker notes

Expectation to audience:
- a basic knowledge of WildFly and basic knowledge of Kubernetes

WildFly and particularly JBoss EAP - as a Red Hat's product based on the WildFly project - focuses on smooth integration to OpenShift. Some of the design decisions is based on that fact.

AGENDA
WILDFLY TO KUBERNETES

DEMO PROJECT

WildFly Quickstart / HelloWorld

- Servlet
- CDI

https://github.com/wildfly/quickstart

WILDFLY CONTAINERS

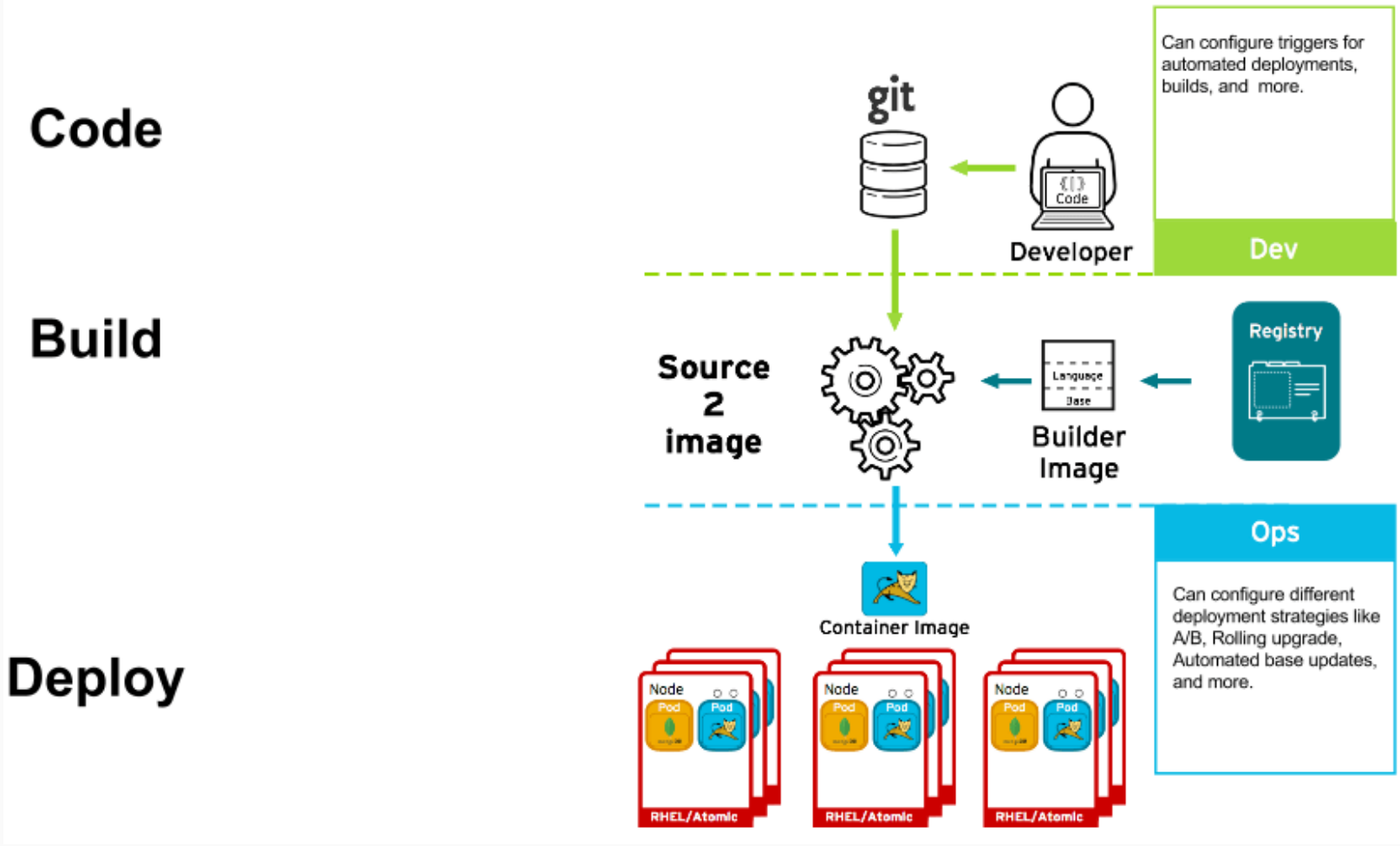
https://quay.io/organization/wildfly

- quay.io/wildfly/wildfly-centos7
- quay.io/wildfly/wildfly-runtime-centos7
- quay.io/wildfly/wildfly-operator

Speaker notes

- wildfly-centos7: s2i image with environment prepared for galleon to work
- wildfly-runtime-centos7: no WFLY included. a bare container with set-up environmental variables that is to be shipped with a container in a chain build
- wildfly-operator: Kubernetes Operator for managing WFLY

S2I



Source: https://blog.openshift.com/save-yourself-from-the-next-glibc/

Speaker notes

- abbreviation for source to image
- goal : building artifacts from sources and injecting them to container
- executing scripts and using plugins to adjust the behaviour

* https://github.com/openshift/source-to-image
* https://docs.openshift.com/aro/creating_images/s2i.html

```
s2i build
  -e GALLEON_PROVISION_DEFAULT_FAT_SERVER=true
  -e MAVEN_OPTS="-Dcom.redhat.xpaas.repo.jbossorg"
  --context-dir helloworld
  --ref 18.0.0.Final
  https://github.com/wildfly/quickstart
  quay.io/wildfly/wildfly-centos7
  helloworld-wildfly-centos7
```

s2i command to build

environmental variables configures how the build goes

directory where build is started

tag/branch

the source code

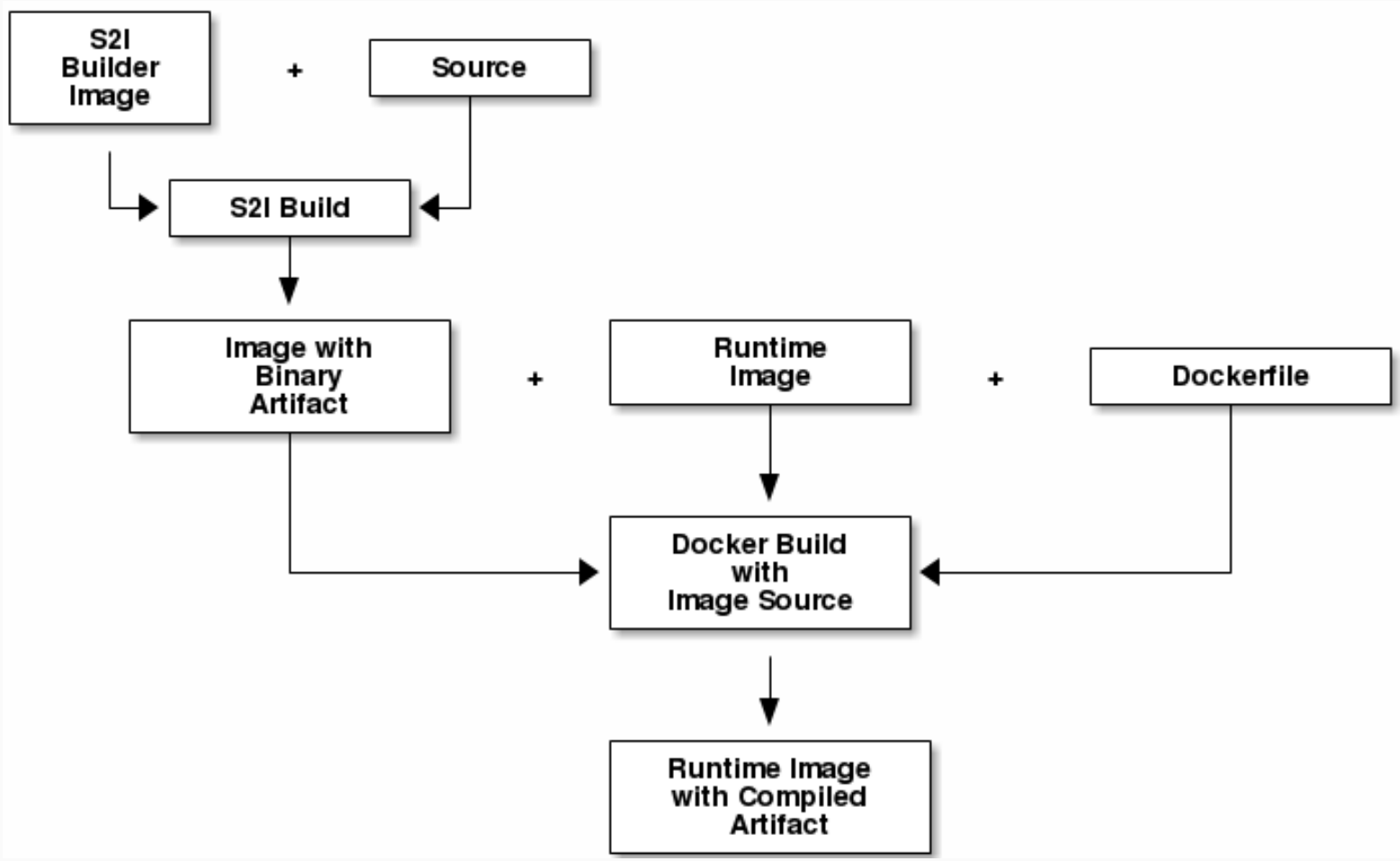
base docker image

result docker image

Speaker notes

* https://github.com/wildfly/wildfly-s2i
* https://github.com/cekit/cekit (CEKit helps to build container images from image definition files with strong focus on modularity and code reuse)
* https://github.com/wildfly/quickstart
* https://github.com/wildfly/quickstart/blob/master/helloworld/src/main/java/org/jboss/as/quickstarts/helloworld/HelloWorld

CHAINING BUILDS



Source: https://blog.openshift.com/chaining-builds/

WHY S2I

- Galleon features available
- OpenShift integration
  - wildfly/wildfly-s2i / wildfly-s2i-chained-build-template.yml

Speaker notes

- OpenShift upstream project is Okd.io

WILDFLY MODULES

```
java
  -jar $JBOSS_HOME/jboss-modules.jar
  -mp $JBOSS_HOME/modules
  org.jboss.as.standalone
```

← java command

↙ jar of the module system

← where the modules reside

← start-up module

Speaker notes

- JBoss Modules is a standalone implementation of a modular (non-hierarchical) class loading and execution environment for Java

* http://jboss-modules.github.io/jboss-modules/manual/
* https://wildfly.org/news/2016/12/12/Jigsaws-Missing-Pieces (Jason Green's critique on Jigsaw project)

GALLEON

```
galleon.sh install ← galleon command to install
  wildfly:current ← WildFly maven repo
  --layers=jaxrs,cdi ← Layers to be generated
  --dir=my-wildfly-server ← Output directory
```

Galleon layers - XML descriptors for WildFly Core build

- wildfly/wildfly-core / core-galleon-pack/src/main/resources/layers/standalone

Speaker notes

- provisioning tool used for WildFly builds

* https://docs.wildfly.org/galleon/ (documetation)
* https://wildfly.org/news/2019/12/17/Ship-your-WildFly-additions-via-Galleon-feature-packs/ (WildFly blogpost)
* https://wildfly.org/news/2019/03/01/Galleon_Openshift/ (Galleon on OpenShift)

DEMO PROJECT

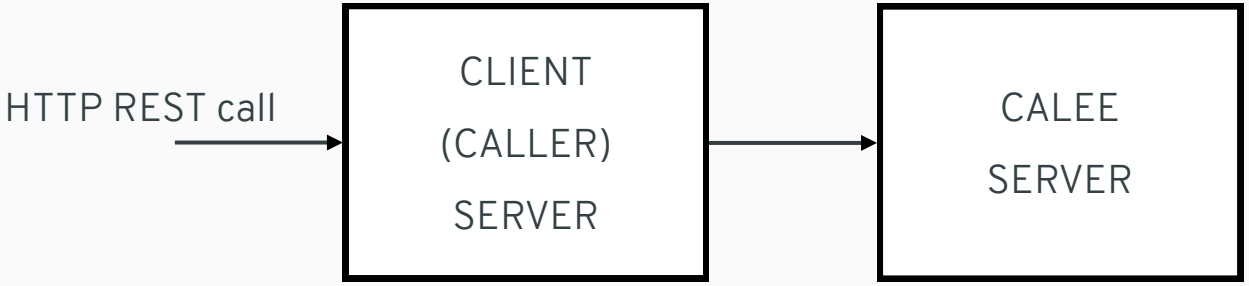
- JAX-RS (HTTP REST)
- EJB
- JPA (Hibernate ORM)
- JMS (Messaging)
- JTA (Transactions)
- Clustering

https://github.com/ochaloup/wildfly-kubernetes-presentation

Speaker notes

Demo is simple but shows the multiple Java EE technologies. We will touch them with concern of the journey to run them in Kubernetes.

DEMO PROJECT



- EJB
- JTA
- JMS

- EJB
- JTA
- JPA
- Clustering

DEBUGGING

- environmental variable **DEBUG**

  kubectl port-forward <pod> 8787:8787

- s2i magic debugging with variable **SCRIPT_DEBUG**

Speaker notes

* https://blog.openshift.com/debugging-java-applications-on-openshift-kubernetes


ENJOY THE REST
OF THE
CONFERENCE