WildFly & Thorntail



About me



"Who":"Jochen Cordes", journey with JBoss and other products "Age": 45, side (they have cookies) and changed "Where": "Neuss", "Status": "Married with children (7w/5m)" "Passion": "Building Open Source solutions with customers", "Job Title": "Solution Architect Middleware"

I'm an Open Source enthusiast since I got in touch with Linux back in 1994 while trying to study Mathematics in Göttingen and got tired to go to the institute or data center to compile C programs for Unix.

In 1999 I got infected by the Java virus and worked as a Java Developer at a startup company which went IPO.

While looking for another application-server I stumbled upon JBoss in 2001 and decided to write my Diploma thesis on Computer Science at the University of Applied Science in Dortmund with and about JBoss in 2004.

I joined a Telecommunications Company in Düsseldorf by then, where I worked in various positions as Software Developer, Tester, Integration Lead, Project Manager and Operation Specialist.

In 2012 I got hired by Red Hat as a Consultant and continued my Middleware journey with JBoss and other products, but got tempted in 2015 by the dark side (they have cookies) and changed into a Pre-Sales role.



Agenda

WildFly

- WildFly, an introduction to the JEE 8 certified Application-Server
- Development with WildFly, Demo
- Why still JEE?
- WildFly as a base for Microservices

Thorntail

- Thorntail, just enough Application-Server
- Development with Thorntail, Demo

Weitere Themen

- Comparison with other platform (Startup-time, memory consumption etc.)
- WildFly and Thorntail in a container



WildFly









http://wildfly.org/



JEE 8 certified Application-Server

- Modular architecture
- On-Demand module loading
- Small footprint, fast startup
- Extensible
- 100% Open Source





Modular Service Container

- Small, lightweight and efficient (233k jar)
- Highly concurrent & scalable state machine
- Various startup modes
 - \circ Active
 - Passive
 - Lazy
 - Never





JBoss Modules

- \$WF_ROOT/jboss-modules.jar (459k)
 - Highly concurrent classloading
 - Fast O(1) dependency resolution
 - \circ $\,$ See only the modules you import $\,$
 - \circ $\,$ No need to break open the jars



java -jar jboss-modules.jar -mp <module-path>
<module-spec>



module.xml

<module name="org.hibernate" xmlns="urn:jboss:module:1.7"> <resources>

<resource-root path="hibernate-core-5.3.6.Final.jar"/>

<resource-root path="hibernate-envers-5.3.6.Final.jar"/>

</resources>

<dependencies>

\$WF_HOME/modules/system/layers/base/org/hibernate/main/module.xml



21:46:57.108 INFO [org.wildfly.security] (ServerService Thread Pool -- 25) ELY00001: WildFly Elytron version 1.6.0.Final 21:46:57,447 INFO [org.jboss.as.controller.management-deprecated] (Controller Boot Thread) WFLYCTL0028: Attribute 'security-realm' in the resource at address '/core-service=management/management-interface=httpinterface' is deprecated, and may be removed in a future version. See the attribute description in the output of the read-resource-description operation to learn more about the deprecation. 21:46:57,473 INFO [org.jboss.as.controller.management-deprecated] (ServerService Thread Pool -- 34) WFLYCTL0028: Attribute 'security-realm' in the resource at address '/subsystem=undertow/server=default-server/ https-listener=https' is deprecated, and may be removed in a future version. See the attribute description in the output of the read-resource-description operation to learn more about the deprecation. 21:46:57,509 INFO [org.jboss.as.server] (Controller Boot Thread) WFLYSRV0039: Creating http management service using socket-binding (management-http) 21:46:57.522 INF0 [org.xnio] (MSC service thread 1-6) XNIO version 3.6.5.Final 21:46:57,527 INFO [org.xnio.nio] (MSC service thread 1-6) XNIO NIO Implementation Version 3.6.5.Final [org.jboss.as.clustering.infinispan] (ServerService Thread Pool -- 48) WFLYCLINF0001: Activating Infinispan subsystem. 21:46:57,551 INF0 21:46:57.578 INFO [org.wildfly.extension.microprofile.config.smallrye. private] (ServerService Thread Pool -- 57) WFLYCONF0001: Activating WildFly MicroProfile Config Subsystem 21:46:57.580 INFO [org.wildfly.extension.io] (ServerService Thread Pool -- 49) WFLYI0001: Worker 'default' has auto-configured to 16 core threads with 128 task threads based on your 8 available processors 21:46:57,580 INF0 [org.jboss.as.jaxrs] (ServerService Thread Pool -- 50) WFLYRS0016: RESTEasy version 3.6.1.Final 21:46:57.582 INFO [org.jboss.as.naming] (ServerService Thread Pool -- 60) WFLYNAM0001: Activating Naming Subsystem 21:46:57,583 WARN [org.jboss.as.txn] (ServerService Thread Pool -- 68) WFLYTX0013: The node-identifier attribute on the /subsystem=transactions is set to the default value. This is a danger for environments run ning multiple servers. Please make sure the attribute value is unique. 21:46:57,590 INFO [org.jboss.as.security] (ServerService Thread Pool -- 66) WFLYSEC0002: Activating Security Subsystem [org.]boss.as.connector.subsystems.datasources] (ServerService Thread Pool -- 41) WFLYJCA0004: Deploving JDBC-compliant driver class org.h2.Driver (version 1.4) 21:46:57,598 INF0 21:46:57,601 INFO [org.jboss.as.security] (MSC service thread 1-2) WFLYSEC0001: Current PicketBox version=5.0.3.Final 21:46:57.600 INFO [org.wildflv.extension.microprofile.opentracing] (ServerService Thread Pool -- 59) WFLYTRACEXT0001: Activating MicroProfile OpenTracing Subsystem 21:46:57,602 INFO [org.wildfly.extension.microprofile.health.smallrye] (ServerService Thread Pool -- 58) WFLYHEALTH0001: Activating Eclipse MicroProfile Health Subsystem 21:46:57.604 INFO [org.jboss.as.isf] (ServerService Thread Pool -- 55) WFLYJSF0007: Activated the following JSF Implementations: [main] 21:46:57,607 INFO [org.jboss.as.webservices] (ServerService Thread Pool -- 70) WFLYWS0002: Activating WebServices Extension 21:46:57.612 INFO [org.jboss.as.connector] (MSC service thread 1-8) WFLYJCA0009: Starting JCA Subsystem (WildFly/IronJacamar 1.4.11.Final) 21:46:57,618 INF0 [org.jboss.remoting] (MSC service thread 1-5) JBoss Remoting version 5.0.8.Final 21:46:57.619 INFO [org.jboss.as.mail.extension] (MSC service thread 1-7) WFLYMAIL0002: Unbound mail session [java:jboss/mail/Default] 21:46:57,636 INFO [org.jboss.as.connector.deployers.jdbc] (MSC service thread 1-1) WFLYJCA0018: Started Driver service with driver-name = h2 21:46:57.649 INF0 [org.iboss.as.naming] (MSC service thread 1-7) WFLYNAM0003: Starting Naming Service 21:46:57,650 INFO [org.wildfly.extension.undertow] (MSC service thread 1-4) WFLYUT0003: Undertow 2.0.13.Final starting [org.jboss.as.mail.extension] (MSC service thread 1-4) WFLYMAIL0001: Bound mail session [java:jboss/mail/Default] 21:46:57,651 INF0 21:46:57,679 INF0 [org.jboss.as.connector.subsystems.datasources] (MSC service thread 1-5) WFLYJCA0010: Unbound data source [java:jboss/datasources/ExampleDS] 21:46:57.740 INFO [org.jboss.as.ejb3] (MSC service thread 1-7) WFLYEJB0482: Strict pool mdb-strict-max-pool is using a max instance size of 32 (per class), which is derived from the number of CPUs on this host. 21:46:57,740 INF0 [org.jboss.as.ejb3] (MSC service thread 1-4) WFLYEJB0481: Strict pool slsb-strict-max-pool is using a max instance size of 128 (per class), which is derived from thread worker pool sizing. 21:46:57.872 INFO [org.wildflv.extension.undertow] (ServerService Thread Pool -- 69) WFLYUT0014: Creating file handler for path '/home/icordes/Software/WildFlv/wildflv-14.0.1.Final/welcome-content' with options [directory-listing: 'false', follow-symlink: 'false', case-sensitive: 'true', safe-symlink-paths: '[]'] 21:46:57.877 INFO [org.wildflv.extension.undertow] (MSC service thread 1-5) WFLYUT0012: Started server default-server. 21:46:57,881 INF0 [org.wildfly.extension.undertow] (MSC service thread 1-5) WFLYUT0018: Host default-host starting 21:46:57,920 INF0 [org.wildfly.extension.undertow] (MSC service thread 1-2) WFLYUT0006: Undertow HTTP listener default listening on 127.0.0.1:8080 21:46:57.954 INFO [org.jboss.as.ejb3] (MSC service thread 1-6) WFLYEJB0493: EJB subsystem suspension complete 21:46:58,006 INFO [org.iboss.as.connector.subsystems.datasources] (MSC service thread 1-7) WFLYJCA0001: Bound data source [java:iboss/datasources/ExampleDS] 21:46:58,086 INF0 [org.jboss.as.patching] (MSC service thread 1-5) WFLYPAT0050: WildFly Full cumulative patch ID is: base, one-off patches include: none 21:46:58,100 WARN [org.jboss.as.domain.management.security] (MSC service thread 1-2) WFLYDM0111: Keystore /home/jcordes/Software/WildFly/wildFly/wildFly-14.0.1.Final/standalone/configuration/application.keystore not fo und, it will be auto generated on first use with a self signed certificate for host localhost 21:46:58,106 INFO [org.jboss.as.server.deployment.scanner] (MSC service thread 1-6) WFLYDS0013: Started FileSystemDeploymentService for directory /home/icordes/Software/WildFly/wildfly-14.0.1.Final/standalone/d eployments 21:46:58,148 INFO [org.wildfly.extension.undertow] (MSC service thread 1-5) WFLYUT0006: Undertow HTTPS listener https listening on 127.0.0.1:8443 21:46:58.194 INFO [org.jboss.ws.common.management] (MSC service thread 1-5) JBWS022052: Starting JBossWS 5.2.3.Final (Apache CXF 3.2.5.jbossorg-1) 21:46:58,236 INF0 [org.iboss.as.server] (Controller Boot Thread) WFLYSRV0212: Resuming server 21:46:58,238 INF0 [org.jboss.as] (Controller Boot Thread) WFLYSRV0060: Http management interface listening on http://127.0.9/1:9990/managem 21:46:58,238 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0051: Admin console listening on http://127.0.0.1:9990 [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: WildFly Full 14.0.1. Final (WildFly Core 6.0.2. Final) started in 2240ms - Started 306 of 527 services (321 services are lazy, passive or on-21:46:58,239 INF0 demand)

WF HOME/bin/standalone.sh









Quickstarts for WildFly 14

```
curl http://download.jboss.org/wildfly/14.0.1.Final/wildfly-14.0.1.Final.zip
-o wildfly-14.0.1.Final.zip
unzip wildfly-14.0.1.Final.zip
cd wildfly-14.0.1.Final
export WF HOME=`pwd`
git clone https://github.com/wildfly/guickstart
git branch
git checkout 14.x
cd kitchensink
mvn clean install
cp target/kitchensink.war $WF HOME/standalone/deployments/
$WF HOME/bin/add-user.sh
$WF HOME/bin/standalone.sh
```



Thorntail





Thorntail offers an innovative approach to packaging and running Java EE applications by packaging them with just enough of the server runtime to "java -jar" your application. It's MicroProfile compatible, too. And, it's all much, much cooler than that ...

Getting Started

📽 Generate Project

The simplest way to get started is by using the project generator. Simply choose the capabilities you need and we'll generate a maven project for you, ready to download and use.

Create a Project

Documentation

View Documentation

The user guide contains much information to get started and to answer more advanced questions.



</> Code Examples

We provide plenty of examples on how to use existing Java EE API's and components to support microservice architectures.

View Examples



https://thorntail.io/



Just enough App Server (JeAS)

- Bundles your deployment (JAR/WAR)
- *Fractions* define available functionality for deployment
- Internal Maven repository with dependencies
- Bootstrap Code
- "Leverage your Java EE expertise and bring your Enterprise Java EE microservices cloud"









- Begins with WildFly / Java EE but focus on modern cloud applications
- Adds non-Java EE cloud native capabilities
 - Netflix OSS, Microprofile, etc.
- Package for deployment
 - Uber Jar (packages your app + what's needed to run it)
 - Hollow Uber Jar (excludes .war) (NOT supported in first RHOAR GA!)



"Right Size" the Runtime

Monolithic architectures

Microservice architectures

















Example on actual resource usage

Runtime ^{[1][2]} (framework)	Boot time server only	Boot time including app deployment	Memory usage without load	Memory usage under load	Measured ^[3] throughput
JBoss EAP (Java EE)	2 - 3 sec	3 sec	40 MB	200 - 400 MB	23K req/sec
JBoss EAP (Spring)	2 - 3 sec	7 sec	40 MB	500 - 700 MB	9K req/sec
JBoss WS/Tomcat (Spring)	0 - 1 sec	8 sec	40 MB	0.5 - 1.5 GB	8K req/sec
Fat JAR (Spring Boot)	N/A	3 sec	30 MB	0.5 - 2.0 GB	11K req/sec
Fat JAR (Thorntail)	1-2 sec	5 sec	30 MB	250 - 350 MB	27K req/sec

Don't believe it? Try it out yourself http://bit.ly/modern-java-runtimes

[1] The microservice is a simple REST application.

[2] All runtimes are using their default settings

[3] The performance test was conducted with ApacheBench using 500K request with 50 users and keep-alive enabled.







Thorntail "pieces" - Fractions

- A tangible unit providing a specific piece of functionality
- Embodied in a maven artifact
- To support the compositional aspect in Thorntail
- Provides the "runtime" capabilities
- Means to configure the system
 - With reasonable defaults
- Means to discover other components (topology)
- Means to alter deployments (e.g. keycloak)
- Can be auto-detected or explicitly declared





Fractions...

- Enable Thorntail subsystems (e.g. Infinispan)
- Integrate additional frameworks/services (e.g. Topology)
- Provide deployments (e.g. Swagger, Jolokia)
- Add API dependencies (e.g. JAX-RS)
- Alter deployments (e.g. SSO)

- Expressed as Maven (GAV) coordinates:
 - o org.wildfly.swarm:<fraction>:<version>
 - o org.wildfly.swarm:undertow:2017.4.0
- 184 fractions currently available
 - \circ 158 stable
 - 26 experimental
- About 80% wrap WildFly related components (Java EE, WF Camel, internal WF)
- More in the pipeline



Fraction Use Cases

- Fractions support explicit and implicit configuration
 - In many cases you won't need to configure anything
 - Or configuration can be externalized
- Fractions can be detected or explicitly declared
 - The most simple case is a <war> project, with just the maven plugin
- All of Java EE is supported in upstream Thorntail
 - JPA, JAX-RS, EJB, JMS, CDI, ...
 - Successfully ran <u>Java EE 7 samples</u> [blog]
 - Successfully ran <u>JBoss EAP 7 Quickstarts [blog</u>]



Building options

- Thorntail Maven Plugin
 - Configurable via options
- Gradle (experimental)
 - "Gradle doesn't get the same attention by the core developers as Maven. If you want to be on the safe side, use Maven. If you are interested in contributing to the Gradle build system, please let us know!"



Tooling

- <u>JBoss Forge add-on</u>
 - IDE Extension for Eclipse/JBDS/IntelliJ IDEA/NetBeans to generate code/scaffolding

```
forge$ project-new --named demo --stack JAVA_EE_7
forge$ wildfly-swarm-setup
forge$ wildfly-swarm-new-test --target-package org.example.rest --named HelloWorldEndpointTest
```

- <u>SwarmTool</u>
 - Standalone jar that is used to wrap an existing JAR or WAR file in a -swarm.jar without requiring any other tooling
- Project Generator
 - Akin to start.spring.io



Tooling - Project Generator

Thorntail				JBOSS.ORG RED HAT	naven-central v2.2.0.Final
	GENERATOR	BLOG	DOCUMENTATION	COMMUNITY	RESOURCES ×
Thorntail Project Generator					
Rightsize your Java EE or Eclipse MicroProfile micros	ervice in a few clicks				
Instructions					
 Choose the dependencies you need Click on the Generate button to download the of Unzip the file in a directory of your choice Run mvn thorntail:run in the unzipped director Go to http://localhost:8080/hello and you shoul Hello from Thorntail! 	<i>lemo.zip</i> file ory Id see the following messa	age:			
com.example					
Artifact ID					
demo		Gener	ate Project		
Dependencies					
JAX-RS, EJB, Transactions, Ribbon, Hibernate Search					Q
Not sure what you are looking for? View all available of	dependencies filtered by :	All	\$		
Selected dependencies					
JAX-RS × CDI Configuration × EJB ×					



[INFO] Starting .war Wed Nov 21 13:31:59 CET 2018 INFO [org.wildfly.swarm.bootstrap] (main) Dependencies not bundled; resolving from M2REPO. 2018-11-21 13:32:01,188 INFO [org.wildfly.swarm] (main) THORN0013: Installed fraction: JAX-RS - STABLE io.thorntail:jaxrs:2.2.1.Final 2018-11-21 13:32:01,195 INFO [org.wildfly.swarm] (main) THORN0013: Installed fraction: Logging - STABLE io.thorntail:logging:2.2.1.Final [org.wildfly.swarm] (main) THORN0013: Installed fraction: Undertow - STABLE io.thorntail:undertow:2.2.1.Final 2018-11-21 13:32:01,196 INFO [org.wildfly.swarm] (main) THORN0013: Installed fraction: Elvtron - STABLE io.thorntail:elytron:2.2.1.Final 2018-11-21 13:32:01,196 INFO [org.jboss.msc] (main) JBoss MSC version 1.2.7.SP1 2018-11-21 13:32:02,522 INFO [org.jboss.as] (MSC service thread 1-7) WFLYSRV0049: Thorntail 2.2.1.Final (WildFly Core 3.0.8.Final) starting 2018-11-21 13:32:02.621 INFO [org.wildfly.swarm] (MSC service thread 1-7) THORN0019: Install MSC service for command line args: [] 2018-11-21 13:32:02,656 INFO 2018-11-21 13:32:03,186 INFO [org.wildfly.security] (ServerService Thread Pool -- 6) ELY00001: WildFly Elytron version 1.1.6.Final 2018-11-21 13:32:03,220 INFO [org.jboss.as.jaxrs] (ServerService Thread Pool -- 14) WFLYRS0016: RESTEasy version 3.0.24.Final 2018-11-21 13:32:03,224 INFO [org.jboss.as.naming] (ServerService Thread Pool -- 13) WFLYNAM0001: Activating Naming Subsystem 2018-11-21 13:32:03,227 INFO [org.xnio] (ServerService Thread Pool -- 15) XNIO version 3.5.4.Final 2018-11-21 13:32:03.236 INFO [org.xnio.nio] (ServerService Thread Pool -- 15) XNIO NIO Implementation Version 3.5.4.Final 2018-11-21 13:32:03,238 INFO [org.jboss.as.security] (ServerService Thread Pool -- 17) WFLYSEC0002: Activating Security Subsystem [org.wildfly.extension.undertow] (MSC service thread 1-7) WFLYUT0003: Undertow 1.4.18.Final starting 2018-11-21 13:32:03,241 INFO org.jboss.as.security] (MSC service thread 1-1) WFLYSEC0001: Current PicketBox version=5.0.2.Final 2018-11-21 13:32:03,244 INFO 2018-11-21 13:32:03,259 INFO [org.jboss.as.naming] (MSC service thread 1-7) WFLYNAM0003: Starting Naming Service 2018-11-21 13:32:03,266 INFO [org.wildfly.extension.io] (ServerService Thread Pool -- 15) WFLY10001: Worker 'default' has auto-configured to 16 core threads with 128 task threads based on your 8 available proce ssors 2018-11-21 13:32:03,360 INFO [org.wildfly.extension.undertow] (MSC service thread 1-1) WFLYUT0012: Started server default-server. 2018-11-21 13:32:03,411 INFO [org.wildfly.extension.undertow] (MSC service thread 1-4) WFLYUT0006: Undertow HTTP listener default listening on [0:0:0:0:0:0:0:0]:8080 2018-11-21 13:32:03,458 INFO [org.jboss.as.server] (Controller Boot Thread) WFLYSRV0212: Resuming server [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: Thorntail 2.2.1. Final (WildFly Core 3.0.8. Final started in 996ms - Started 97 of 103 services (16 services are lazy, passive or 2018-11-21 13:32:03,461 INF0 on-demand) 2018-11-21 13:32:03,654 INFO [org.wildfly.swarm.runtime.deployer] (main) deploying demo.war 2018-11-21 13:32:03.676 INFO [org.jboss.as.server.deployment] (MSC service thread 1-4) WFLYSRV0027: Starting deployment of "demo.war" (runtime-name: "demo.war") 2018-11-21 13:32:04.130 INFO [org.wildfly.extension.undertow] (MSC service thread 1-8) WFLYUT0018: Host default-host starting [org.jboss.resteasy.resteasy jaxrs.i18n] (ServerService Thread Pool -- 6) RESTEASY002225: Deploying javax.ws.rs.core.Application: class com.example.demo.rest.RestApplication 2018-11-21 13:32:04,313 INFO 2018-11-21 13:32:04,330 INFO [org.wildfly.extension.undertow] (ServerService Thread Pool -- 6) WFLYUT0021: Registered web context: '/' for server 'default-server' 2018-11-21 13:32:04,373 INFO [org.jboss.as.server] (main) WFLYSRV0010: Deployed "demo.war" (runtime-name : "demo.war") 2018-11-21 13:32:04,386 INFO [org.wildfly.swarm] (main) THORN99999: Thorntail is Ready

unzip demo.zip

cd demo

mvn thorntail:run



Bootstrapping

- Akin to Spring Boot's @SpringBootApplication •
- Not needed in many cases •
- Provides advanced startup configuration API •

```
public class MyMain {
    public static void main(String...args) {
        // Instantiate the container
        Swarm swarm = new Swarm();
        // Create one or more deployments
        JAXRSArchive deployment = ShrinkWrap.create(JAXRSArchive.class);
        . . .
        // Add resource to deployment
        deployment.addClass(MyResource.class);
        swarm.start();
        swarm.deploy(deployment);
    }
```



}

Uberjar anatomy (for the curious)





Eclipse MicroProfile

http://microprofile.io





A community of individuals, organizations, and vendors collaborating within an open source (Eclipse) project to accelerate the adoption of microservices within the Enterprise Java community





Eclipse MicroProfile and Thorntail

- Thorntail is the Red Hat implementation of Eclipse MicroProfile
- Thorntail will facilitate innovation of future Eclipse MicroProfile features
 - Health Check proposal derived from monitor fraction in Thorntail
- Some Eclipse MicroProfile features may be pushed further upstream into WildFly
 - Provides alignment between WildFly and Thorntail on the same feature
 - Config API will likely be implemented in such a fashion





MicroProfile 1.0

- **Released September 2016** •
- Supported as part of Red Hat OpenShift Application Runtimes (RHOAR) •
- Contained:
 - CDI 1.2
 - **JAX-RS 2.0.1** 0
 - **JSON** Processing 1.0 0





MicroProfile 1.1





Security: JWT Token Exchange 1.0



Health Check 1.0







- Thorntail Status:
 - **Configuration** Implementation being worked on at present 0
 - Health Check Align with `monitor` fraction when released 0
 - **Security** May just require validation with `keycloak` fraction 0
 - Fault Tolerance No implementation commenced 0
- Once 1.1 released, will be available in Thorntail community release some months after •



Using MicroProfile in Thorntail

<dependencies>

<!-- Java EE 7 dependency -->

<dependency>

<proupId>javax</proupId>

<artifactId>javaee-api</artifactId>

<version>7.0</version>

<scope>provided</scope>

</dependency>

<!-- WildFly Swarm Fractions -->

<dependency>

<groupId>org.wildfly.swarm</groupId>

<artifactId>microprofile</artifactId>

</dependency>

</dependencies>



Using MicroProfile in Thorntail





DEPLOYING TO OPENSHIFT



Fabric8 Maven Plugin

- Developed as part of the Fabric8 project (the upstream of openshift.io)
- Generates complete Kubernetes resource objects from scratch, or
- *Enriches* resource object fragments based on Maven settings and autodetected opinionated defaults
- Adds those objects to Kubernetes/OpenShift via kubernetes API
- Can generate ReplicationController/Services/Pods/Builds/Deployments





Fabric8 Maven Plugin - Enrichment

apiVersion: v1 kind: Route spec: port: targetPort: 8080

fabric8:deploy

kind: Route metadata: annotations: openshift.io/host.generated: "true" creationTimestamp: 2017-07-26T21:18:54Z labels: group: com.redhat.coolstore project: inventory provider: fabric8 version: 1.0.0-SNAPSHOT name: inventory namespace: lab3 resourceVersion: "37580" selfLink: /oapi/v1/namespaces/lab3/routes/inventory uid: 074eaaa7-7248-11e7-8265-f636c6486222 spec: host: inventory-lab3.apps.127.0.0.1.nip.io port: targetPort: 8080 to: kind: Service name: inventory weight: 100 wildcardPolicy: None

apiVersion: v1

mvn io.fabric8:fabric8-maven-plugin:3.5.41:resource mvn io.fabric8:fabric8-maven-plugin:3.5.41:deploy





OpenShift Support

- topology-openshift
 - OpenShift integration for WF Swarm Topology, our service registry abstraction
 - topology.js in `topology-webapp` to call Kubernetes services by name
- monitor
 - Health Check endpoints
 - fabric8-maven-plugin configuration to provide Kubernetes Readiness and Liveness checks
- project-defaults.yml loaded from Kubernetes ConfigMap





THANK YOU



plus.google.com/+RedHat

in linkedin.com/company/red-hat



f

facebook.com/redhatinc



twitter.com/RedHat