



RED HAT
FORUM
Europe, Middle East & Africa



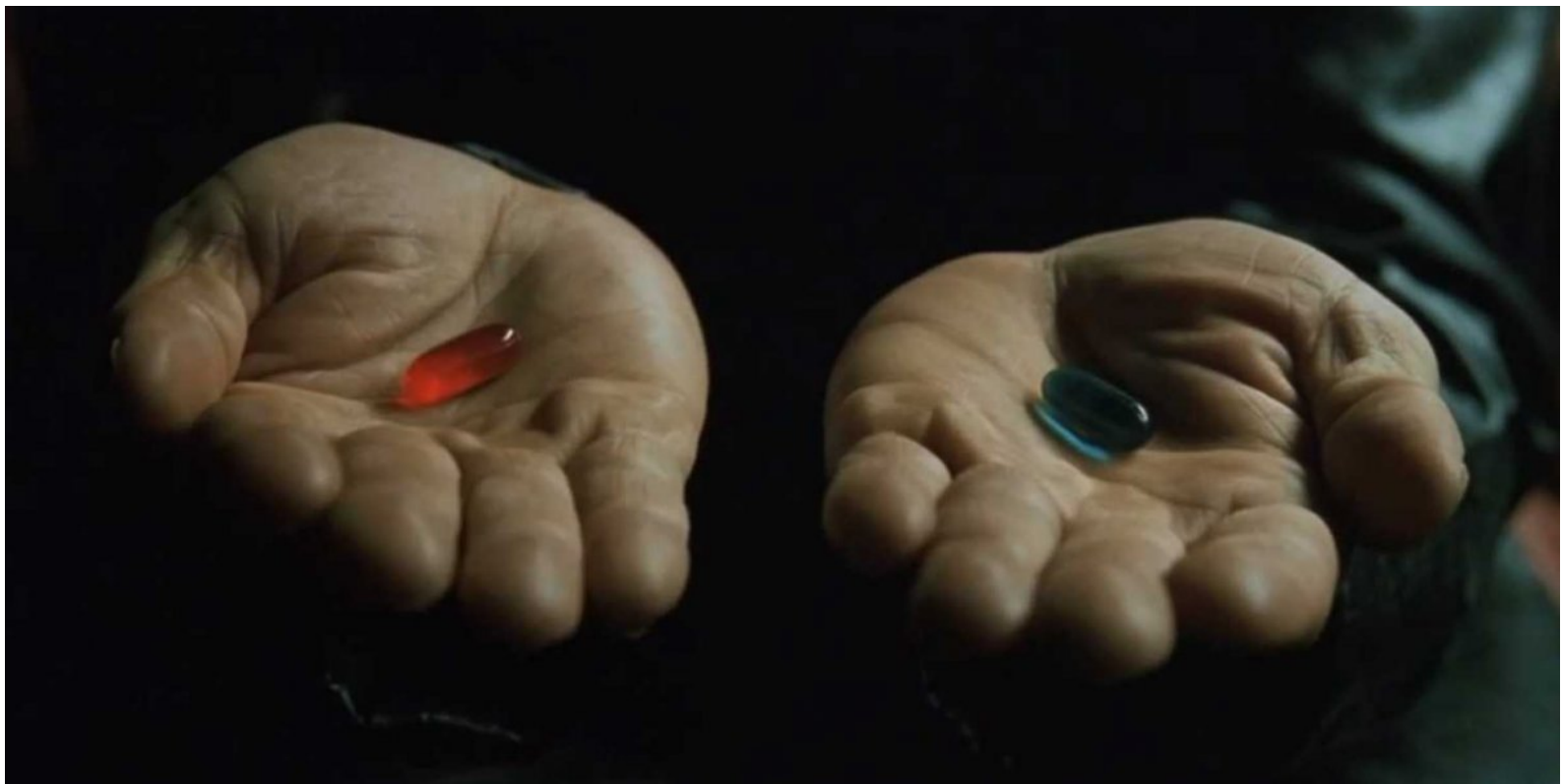
TURN YOUR JAVA EE MONOLITHS INTO MICROSERVICES WITH WILDFLY SWARM

DIMITRIS ANDREADIS

SENIOR ENGINEERING MANAGER
JBoss EAP / WILDFLY

RED HAT

Once upon a time*



Agenda

- Thinking Microservices
- WildFly Architecture
- WildFly Swarm Concepts
- Getting Started with Swarm
- Advanced Configuration Options

Let's talk Microservices

What are Microservices?



What are Microservices?

- A bit like SOA (but smaller?)
- Written in any cool programming language
- By (smaller?) two-pizza teams
- Who break monoliths into (small?) services
- That encapsulate fully some (small?) vertical application aspect
- And run them independently in their own (smaller?) containers

Did I say small?

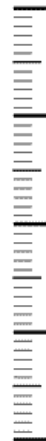
Why do Microservices?

- 1) Because everyone else does it
- 2) Because you have to go internet scale
- 3) Because you want to deliver fast

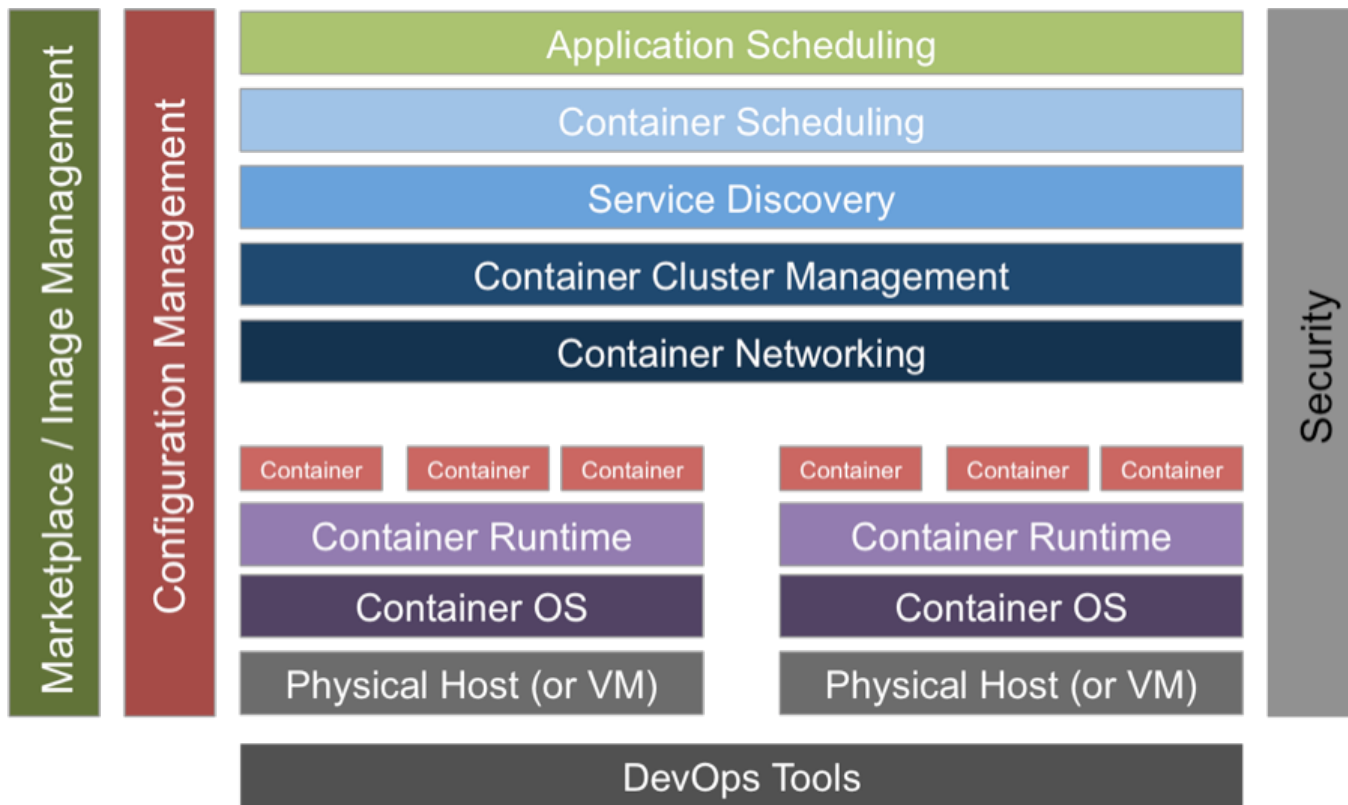
You Must Be This Tall*

- Self-Service, On Demand, Elastic Infrastructure as code
- Deployment Independence
- Full stack Ownership
- Dev vs Ops
- Automation & Immutability
- CI/CD pipeline
- Agile Delivery

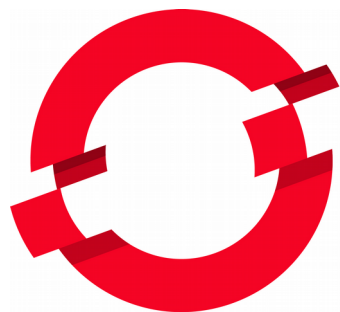
**YOU MUST
BE THIS TALL
TO RIDE
THIS RIDE**



Simpler services but more complex environments



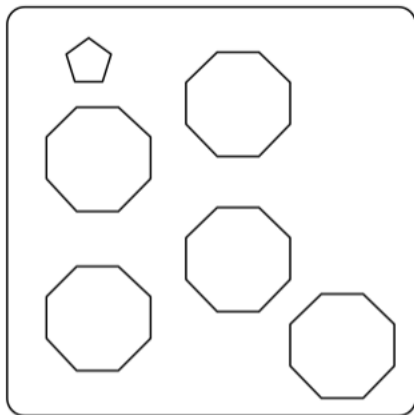
*“Rolling out Microservices
requires some sort of a
Distributed Application Server” **



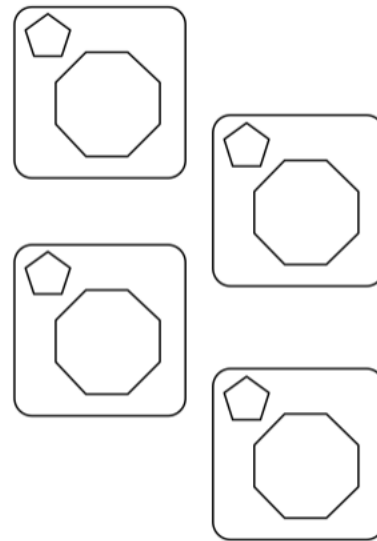
RED HAT[®]
OPENSIFT
Container Platform

“Right Size” the Runtime

Monolithic architectures

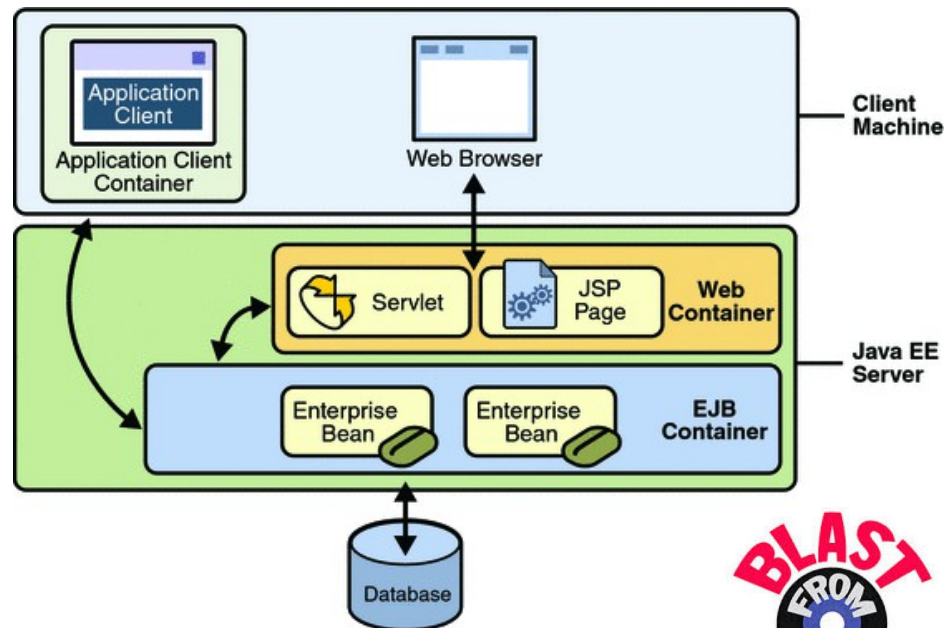


Microservice architectures



JBoss “Innovation” ca. 2001

- Collapse the EJB and Servlet layer in a single VM to avoid **x100** slower remote calls!
- No need to actively **code for remote call failure** either :)



**BLAST
FROM
THE
PAST**

A tiny bit of WildFly Architecture

WildFly* pre-History

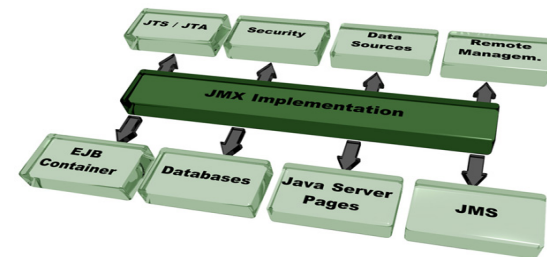


JBoss **Application Server 7**



Kernel Taxonomy

- JBoss AS 2.x, 3.x, 4.x
 - JMX MicroKernel
- JBoss AS 5.x, 6.x
 - JBoss MicroContainer
- AS7.x, WildFly 8.x, 9.x, 10.x
 - JBoss Modules & Modular Service Container




```
java:jboss/datasources/ExampleDS]
16:26:35,949 WARN [org.jboss.as.domain.management.security] (MSC service thread 1-1) WFLYDM0111: Keystore /Users/dimitris/wildfly/wildfly-10.1.0.Final/standalone/configuration/application.keystore not found, it will be auto generated on first use with a self signed certificate for host localhost
16:26:36,132 INFO [org.jboss.as.server.deployment.scanner] (MSC service thread 1-4) WFLYDS0013: Started FileSystemDeploymentService for directory /Users/dimitris/wildfly/wildfly-10.1.0.Final/standalone/deployments
16:26:36,391 INFO [org.infinispan.factories.GlobalComponentRegistry] (MSC service thread 1-4) ISPN000128: Infinispan version: Infinispan 'Chakra' 8.2.4.Final
16:26:36,387 INFO [org.wildfly.extension.undertow] (MSC service thread 1-1) WFLYUT0006: Undertow HTTPS listener https listening on 127.0.0.1:8443
16:26:36,410 INFO [org.infinispan.configuration.cache.EvictionConfigurationBuilder] (ServerService Thread Pool -- 58) ISPN000152: Passivation configured without an eviction policy being selected. Only manually evicted entities will be passivated.
16:26:36,410 INFO [org.infinispan.configuration.cache.EvictionConfigurationBuilder] (ServerService Thread Pool -- 64) ISPN000152: Passivation configured without an eviction policy being selected. Only manually evicted entities will be passivated.
16:26:36,410 INFO [org.infinispan.configuration.cache.EvictionConfigurationBuilder] (ServerService Thread Pool -- 62) ISPN000152: Passivation configured without an eviction policy being selected. Only manually evicted entities will be passivated.
16:26:36,411 INFO [org.infinispan.configuration.cache.EvictionConfigurationBuilder] (ServerService Thread Pool -- 58) ISPN000152: Passivation configured without an eviction policy being selected. Only manually evicted entities will be passivated.
16:26:36,412 INFO [org.infinispan.configuration.cache.EvictionConfigurationBuilder] (ServerService Thread Pool -- 64) ISPN000152: Passivation configured without an eviction policy being selected. Only manually evicted entities will be passivated.
16:26:36,412 INFO [org.infinispan.configuration.cache.EvictionConfigurationBuilder] (ServerService Thread Pool -- 62) ISPN000152: Passivation configured without an eviction policy being selected. Only manually evicted entities will be passivated.
16:26:36,498 INFO [org.jboss.ws.common.management] (MSC service thread 1-1) JBWS022052: Starting JBossWS 5.1.5.Final (Apache CXF 3.1.6)
16:26:36,639 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0060: Http management interface listening on http://127.0.0.1:9990/management
16:26:36,640 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0051: Admin console listening on http://127.0.0.1:9990
16:26:36,640 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: WildFly Full 10.1.0.Final (WildFly Core 2.2.0.Final) started in 3269ms - Started 331 of 577 services (393 services are lazy, passive or on-demand)
```

```
<server xmlns="urn:jboss:domain:4.1">
```

```
<extensions>
```

```
<extension module="org.jboss.as.clustering.infinispan"/>
<extension module="org.jboss.as.connector"/>
<extension module="org.jboss.as.deployment-scanner"/>
<extension module="org.jboss.as.ee"/>
<extension module="org.jboss.as.ejb3"/>
<extension module="org.jboss.as.jaxrs"/>
<extension module="org.jboss.as.jdr"/>
<extension module="org.jboss.as.jmx"/>
<extension module="org.jboss.as.jpa"/>
<extension module="org.jboss.as.jsf"/>
<extension module="org.jboss.as.logging"/>
<extension module="org.jboss.as.mail"/>
<extension module="org.jboss.as.naming"/>
<extension module="org.jboss.as.pojo"/>
<extension module="org.jboss.as.remoting"/>
<extension module="org.jboss.as.sar"/>
<extension module="org.jboss.as.security"/>
<extension module="org.jboss.as.transactions"/>
<extension module="org.jboss.as.webservices"/>
<extension module="org.jboss.as.weld"/>
<extension module="org.wildfly.extension.batch.jberet"/>
<extension module="org.wildfly.extension.bean-validation"/>
<extension module="org.wildfly.extension.io"/>
<extension module="org.wildfly.extension.request-controller"/>
<extension module="org.wildfly.extension.security.manager"/>
<extension module="org.wildfly.extension.undertow"/>
```

```
</extensions>
```

Java EE7 Full & Web
distribution (134 Mb)

standalone/
configuration/
configuration.xml

```
<server xmlns="urn:jboss:domain:4.2">
  <extensions>
    <extension module="org.jboss.as.deployment-scanner"/>
    <extension module="org.jboss.as.ear"/>
    <extension module="org.jboss.as.jmx"/>
    <extension module="org.jboss.as.logging"/>
    <extension module="org.jboss.as.naming"/>
    <extension module="org.jboss.as.security"/>
    <extension module="org.wildfly.extension.io"/>
    <extension module="org.wildfly.extension.request-controller"/>
    <extension module="org.wildfly.extension.security.manager"/>
    <extension module="org.wildfly.extension.undertow"/>
  </extensions>
```

Servlet-Only
distribution (28 Mb)

standalone/
configuration/
configuration.xml


```
16:54:21,369 INFO [org.jboss.remoting] (MSC service thread 1-4) JBoss Remoting version 4.0.21.Final
16:54:21,377 INFO [org.wildfly.extension.io] (ServerService Thread Pool -- 15) WFLYIO0001: Worker 'default' has auto-configured to 8 core threads with 64 task threads based on your 4 available processors
16:54:21,387 INFO [org.jboss.as.naming] (ServerService Thread Pool -- 17) WFLYNAM0001: Activating Naming Subsystem
16:54:21,393 INFO [org.jboss.as.security] (ServerService Thread Pool -- 19) WFLYSEC0002: Activating Security Subsystem
16:54:21,395 INFO [org.jboss.as.security] (MSC service thread 1-6) WFLYSEC0001: Current PicketBox version=4.9.6.Final
16:54:21,403 INFO [org.jboss.as.naming] (MSC service thread 1-8) WFLYNAM0003: Starting Naming Service
16:54:21,436 INFO [org.wildfly.extension.undertow] (MSC service thread 1-7) WFLYUT0003: Undertow 1.4.0.Final starting
16:54:21,472 INFO [org.wildfly.extension.undertow] (ServerService Thread Pool -- 21) WFLYUT0014: Creating file handler for path '/Users/dimitris/wildfly/wildfly-servlet-10.1.0.Final/welcome-content' with options [directory-listing: 'false', follow-symlink: 'false', case-sensitive: 'true', safe-symlink-paths: '[]']
16:54:21,478 INFO [org.wildfly.extension.undertow] (MSC service thread 1-1) WFLYUT0012: Started server default-server.
16:54:21,479 INFO [org.wildfly.extension.undertow] (MSC service thread 1-7) WFLYUT0018: Host default-host starting
16:54:21,541 INFO [org.wildfly.extension.undertow] (MSC service thread 1-6) WFLYUT0006: Undertow HTTP listener default listening on 127.0.0.1:8080
16:54:21,646 WARN [org.jboss.as.domain.management.security] (MSC service thread 1-1) WFLYDM0111: Keystore /Users/dimitris/wildfly/wildfly-servlet-10.1.0.Final/standalone/configuration/application.keystore not found, it will be auto generated on first use with a self signed certificate for host localhost
16:54:21,655 WARN [org.jboss.as.domain.http.api.undertow] (MSC service thread 1-4) WFLYDMHTTP0003: Unable to load console module for slot main, disabling console
16:54:21,660 INFO [org.jboss.as.server.deployment.scanner] (MSC service thread 1-8) WFLYDS0013: Started FileSystemDeploymentService for directory /Users/dimitris/wildfly/wildfly-servlet-10.1.0.Final/standalone/deployments
16:54:21,744 INFO [org.wildfly.extension.undertow] (MSC service thread 1-6) WFLYUT0006: Undertow HTTPS listener https listening on 127.0.0.1:8443
16:54:21,852 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0060: Http management interface listening on http://127.0.0.1:9990/management
16:54:21,852 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0051: Admin console listening on http://127.0.0.1:9990
16:54:21,853 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: WildFly Servlet 10.1.0.Final (WildFly Core 2.2.0.Final) started in 1864ms - Started 109 of 114 services (34 services are lazy, passive or on-demand)
```



WildFly

*"the Swiss Army
Knife of Runtimes"*

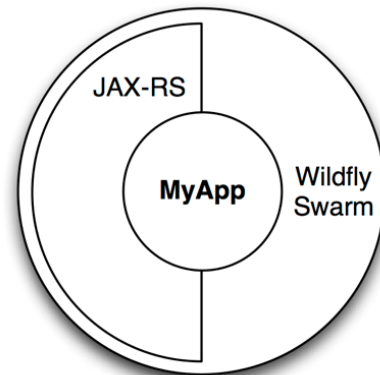
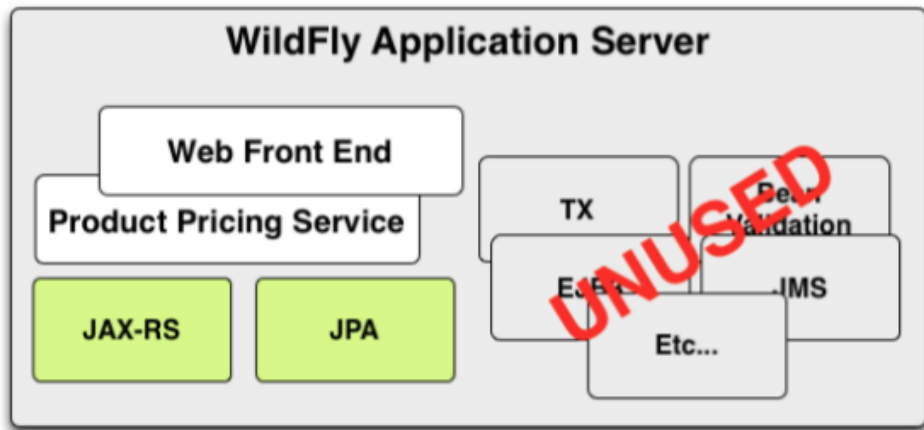
Welcome to WildFly Swarm

What is WildFly Swarm?



- WildFly Broken Apart
 - Decompose and reconstruct WildFly around your application
 - Just enough app-server for your Microservice packaged into an UberJar
 - With additional functionality outside Java EE like NetflixOSS libraries, etc.
- Part of the JBoss / Red Hat ecosystem

Rightsize your Runtime around your App



Plus a lot more components!

- Keycloak (SSO)
- Hystrix (Circuit Breaker)
- Ribbon (client side LB)
- Vert.x (reactive)
- Undertow (web server)
- Camel (integration)
- Hashicorp Consul (discovery)
- Swagger (service contracts)
- LogStash (log aggregation)
- and more...

Why WildFly Swarm?

- Leverage your Java EE expertise
- Build on mature technologies and a scalable runtime
- Be part of the future of Java EE and help shape it for the Cloud

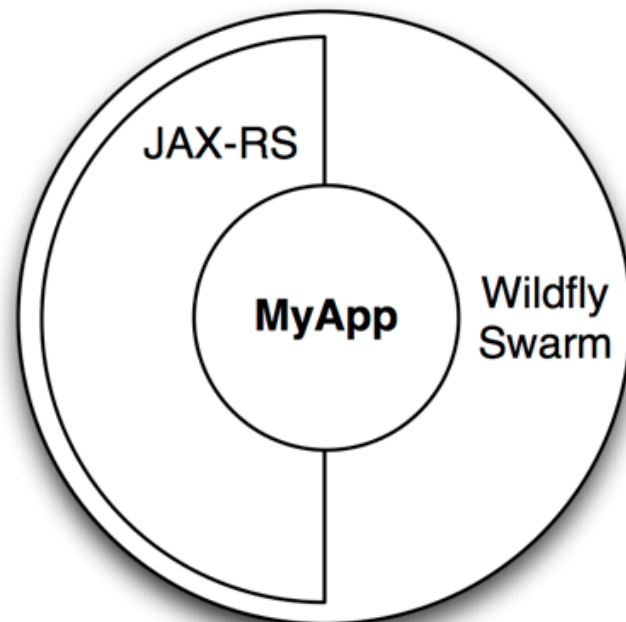
MicroProfile.io

Optimizing Enterprise Java for a microservices architecture



Self-contained (Uber) Jar

- bundles your application,
 - the *Fractions* to support it,
 - an internal Maven repository with the dependencies,
 - and Bootstrap code.
-
- There is also the notion of a *Hollow* launch-pad type of jar.



Concept of a Fraction

- WildFly Swarm compositional units
- Providing specific runtime capabilities
- And different means to configure them
 - with reasonable default

Fractions

- Enable WildFly 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)

Fractions (cont.)

- Expressed as Maven (GAV) coordinates:
 - `org.wildfly.swarm:<fraction>:<version>`, e.g
 - `org.wildfly.swarm:undertow:2017.09.4`
- 169 fractions currently available
 - 148 stable
 - 15 unstable
 - 6 *experimental*
- More in the pipeline

Where do I start?

<http://wildfly-swarm.io/generator/>

WildFly Swarm Project Generator

Rightsize your Java EE microservice in a few clicks

Instructions

1. Choose the dependencies you need
2. Click on the Generate button to download the *demo.zip* file
3. Unzip the file in a directory of your choice
4. Run `mvn wildfly-swarm:run` in the unzipped directory
5. Go to `http://localhost:8080/hello` and you should see the following message:

Hello from WildFly Swarm!

Group ID

Artifact ID

Generate Project

Dependencies



Not sure what you are looking for? [View all available dependencies](#) filtered by :

Selected dependencies

MicroProfile

demo.zip (1kb)

```
|— pom.xml
|
|— src
|   |
|   |— main
|       |
|       |— java
|           |
|           |— com
|               |
|               |— example
|                   |
|                   |— demo
|                       |
|                       |— rest
|                           |
|                           |— HelloWorldEndpoint.java
```

HelloWorldEndpoint.java

```
package com.example.demo.rest;

import javax.ws.rs.Path;

@Path("/hello")
public class HelloWorldEndpoint {

    @GET
    @Produces("text/plain")
    public Response doGet() {
        return Response.ok("Hello from WildFly Swarm!").build()
    }
}
```

How is WildFly Swarm enabled?

```
<plugin>
  <groupId>org.wildfly.swarm</groupId>
  <artifactId>wildfly-swarm-plugin</artifactId>
  <version>${version.wildfly.swarm}</version>
  <executions>
    <execution>
      <goals>
        <goal>package</goal>
      </goals>
    </execution>
  </executions>
</plugin>
```

```
<dependencies>
  <!-- Java EE 7 dependency -->
  <dependency>
    <groupId>javax</groupId>
    <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>
```

Build

- mvn package



demo-swarm.jar

```
...  
_bootstrap/demo.war  
m2repo/<fractions & modules jars>  
modules/<bootstrap-modules.xml  
<bootstrap-classes>  
...
```

Run

- `java -jar demo-swarm.jar`
- `mvn wildfly-swarm:run`
- IDE > Run o.w.s.Swarm
- IDE > Run ... MyMain

java -jar target/demo-swarm.jar

```
[dimitris@ares ~/dev/src/demo $ java -jar target/demo-swarm.jar
2017-05-13 20:35:44,103 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: Logging - STABLE
org.wildfly.swarm:logging:2017.5.0
2017-05-13 20:35:44,117 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: JAX-RS with JAXB - STABLE
org.wildfly.swarm:jaxrs-jaxb:2017.5.0
2017-05-13 20:35:44,117 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: Bean Validation - STABLE
org.wildfly.swarm:bean-validation:2017.5.0
2017-05-13 20:35:44,118 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: CDI Configuration - STABLE
org.wildfly.swarm:cdi-config:2017.5.0
2017-05-13 20:35:44,118 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: JAX-RS with JSON-P - STABLE
org.wildfly.swarm:jaxrs-jsonp:2017.5.0
2017-05-13 20:35:44,118 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: Transactions - STABLE
org.wildfly.swarm:transactions:2017.5.0
2017-05-13 20:35:44,119 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: CDI - STABLE
org.wildfly.swarm:cdi:2017.5.0
2017-05-13 20:35:44,119 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: MicroProfile - STABLE
org.wildfly.swarm:microprofile:2017.5.0
2017-05-13 20:35:44,120 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: Undertow - STABLE
org.wildfly.swarm:undertow:2017.5.0
2017-05-13 20:35:44,121 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: JAX-RS - STABLE
org.wildfly.swarm:jaxrs:2017.5.0
2017-05-13 20:35:46,883 INFO [org.jboss.msc] (main) JBoss MSC version 1.2.6.Final
2017-05-13 20:35:46,981 INFO [org.jboss.as] (MSC service thread 1-6) WFLYSRV0049: WildFly Swarm 2017.5.0 (WildFly Core 2.2.1.Final)
starting
```

```
2017-05-13 20:42:09,876 WARN [org.jboss.as.txn] (ServerService Thread Pool -- 18) WFLYTX0013: Node identifier property is set to the
default value. Please make sure it is unique.
2017-05-13 20:42:09,937 INFO [org.wildfly.extension.io] (ServerService Thread Pool -- 21) WFLYIO001: Worker 'default' has auto-confi
gured to 8 core threads with 64 task threads based on your 4 available processors
2017-05-13 20:42:09,941 INFO [org.jboss.as.naming] (MSC service thread 1-1) WFLYNAM0003: Starting Naming Service
2017-05-13 20:42:10,012 INFO [org.wildfly.extension.undertow] (MSC service thread 1-7) WFLYUT0003: Undertow 1.4.11.Final starting
2017-05-13 20:42:10,083 INFO [org.xnio] (MSC service thread 1-2) XNIO version 3.4.3.Final
2017-05-13 20:42:10,105 INFO [org.xnio.nio] (MSC service thread 1-2) XNIO NIO Implementation Version 3.4.3.Final
2017-05-13 20:42:10,157 INFO [org.wildfly.extension.undertow] (MSC service thread 1-4) WFLYUT0012: Started server default-server.
2017-05-13 20:42:10,234 INFO [org.wildfly.extension.undertow] (MSC service thread 1-4) WFLYUT0006: Undertow HTTP listener default li
stening on [0:0:0:0:0:0:0:0]:8080
2017-05-13 20:42:10,345 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: WildFly Swarm 2017.5.0 (WildFly Core 2.2.1.Final)
started in 1837ms - started 88 of 97 services (18 services are lazy, passive or on-demand)
2017-05-13 20:42:10,597 INFO [org.wildfly.swarm.runtime.deployer] (main) deploying demo.war
2017-05-13 20:42:10,630 INFO [org.jboss.as.server.deployment] (MSC service thread 1-1) WFLYSRV0027: Starting deployment of "demo.war
" (runtime-name: "demo.war")
2017-05-13 20:42:10,866 WARN [org.jboss.as.dependency.private] (MSC service thread 1-1) WFLYSRV0018: Deployment "deployment.demo.war
" is using a private module ("org.jboss.jts:main") which may be changed or removed in future versions without notice.
2017-05-13 20:42:11,015 INFO [org.jboss.weld.deployer] (MSC service thread 1-3) WFLYWELD0003: Processing weld deployment demo.war
2017-05-13 20:42:11,348 INFO [org.hibernate.validator.internal.util.Version] (MSC service thread 1-3) HV000001: Hibernate Validator
5.2.4.Final
2017-05-13 20:42:11,534 INFO [org.jboss.weld.Version] (MSC service thread 1-4) WELD-000900: 2.3.5 (Final)
2017-05-13 20:42:11,562 INFO [org.wildfly.extension.undertow] (MSC service thread 1-1) WFLYUT0018: Host default-host starting
2017-05-13 20:42:12,356 INFO [org.jboss.resteasy.resteasy_jaxrs.i18n] (ServerService Thread Pool -- 9) RESTEASY002225: Deploying jav
ax.ws.rs.core.Application: class org.wildfly.swarm.generated.WildFlySwarmDefaultJAXRSApplication
2017-05-13 20:42:12,640 INFO [org.wildfly.extension.undertow] (ServerService Thread Pool -- 9) WFLYUT0021: Registered web context: /
2017-05-13 20:42:12,679 INFO [org.jboss.as.server] (main) WFLYSRV0010: Deployed "demo.war" (runtime-name : "demo.war")
2017-05-13 20:42:12,681 INFO [org.wildfly.swarm] (main) WFSWARM99999: WildFly Swarm is Ready
```


Auto-detecting Fractions

- Absence of dependencies triggers *fractionDetectMode*

```
<!-- <dependency> -->
<!-- <groupId>org.wildfly.swarm</groupId> -->
<!-- <artifactId>microprofile</artifactId> -->
<!-- </dependency> -->
</dependencies>
```

```
[INFO] --- wildfly-swarm-plugin:2017.5.0:package (default) @ demo ---
[INFO] Scanning for needed WildFly Swarm fractions with mode: when_missing
[INFO] Detected fractions: jaxrs:2017.5.0
[INFO] Adding fractions: container:2017.5.0, jaxrs-cdi:2017.5.0, jaxrs:2017.5.0,
logging:2017.5.0, security:2017.5.0, undertow:2017.5.0
Resolving 46 out of 206 artifacts
[INFO] Repackaging .war: /Users/dimitris/dev/src/demo/target/demo.war
```

JAXRS fraction selected

```
dimitris@Ares ~/dev/src/demo $ java -jar target/demo-swarm.jar
2017-05-13 20:58:20,993 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: Logging - STABLE
org.wildfly.swarm:logging:2017.5.0
2017-05-13 20:58:21,002 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: JAX-RS - STABLE
org.wildfly.swarm:jaxrs:2017.5.0
2017-05-13 20:58:21,003 INFO [org.wildfly.swarm] (main) WFSWARM0013: Installed fraction: Undertow - STABLE
org.wildfly.swarm:undertow:2017.5.0
```

Bean Validation, CDI, CDI-Config, Transactions, JAX-RS w.
JAXB, JAX-RS w. JSON-P, Microprofile fractions removed

```
2017-05-13 20:58:24,509 INFO [org.wildfly.extension.undertow] (MSC service thread 1-6) WFLYUT0006: Undertow HTTP listener default
listening on [0:0:0:0:0:0:0:0]:8080
2017-05-13 20:58:24,630 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: WildFly Swarm 2017.5.0 (WildFly Core 2.2.1.Final
) started in 1413ms - started 70 of 77 services (14 services are lazy, passive or on-demand)
2017-05-13 20:58:24,867 INFO [org.wildfly.swarm.runtime.deployer] (main) deploying demo.war
2017-05-13 20:58:24,904 INFO [org.jboss.as.server.deployment] (MSC service thread 1-1) WFLYSRV0027: Starting deployment of "demo.w
ar" (runtime-name: "demo.war")
```

How Fractions are configured?

- Default configuration with sensible defaults
- System properties to override common settings
- standalone.xml or a fragment of it can override settings
 - or -
- Take control of main() and use the fluent Java API
 - Covers all WildFly subsystems
 - Generated from the WF configuration schema

Taking control of main()

```
package com.example.demo;

import org.wildfly.swarm.Swarm;
import org.wildfly.swarm.jaxrs.JAXRSFraction;

public class Main {
    public static void main(String args[]) throws Exception {
        Swarm swarm = new Swarm();

        // create and configure fractions
        swarm.fraction(new JAXRSFraction());

        // start the container and deploy the app
        swarm.start().deploy();
    }
}
```

```
<plugin>
  <groupId>org.wildfly.swarm</groupId>
  <artifactId>wildfly-swarm-plugin</artifactId>
  <version>${version.wildfly.swarm}</version>
  <configuration>
    <mainClass>com.example.demo.Main</mainClass>
  </configuration>
</plugin>
```

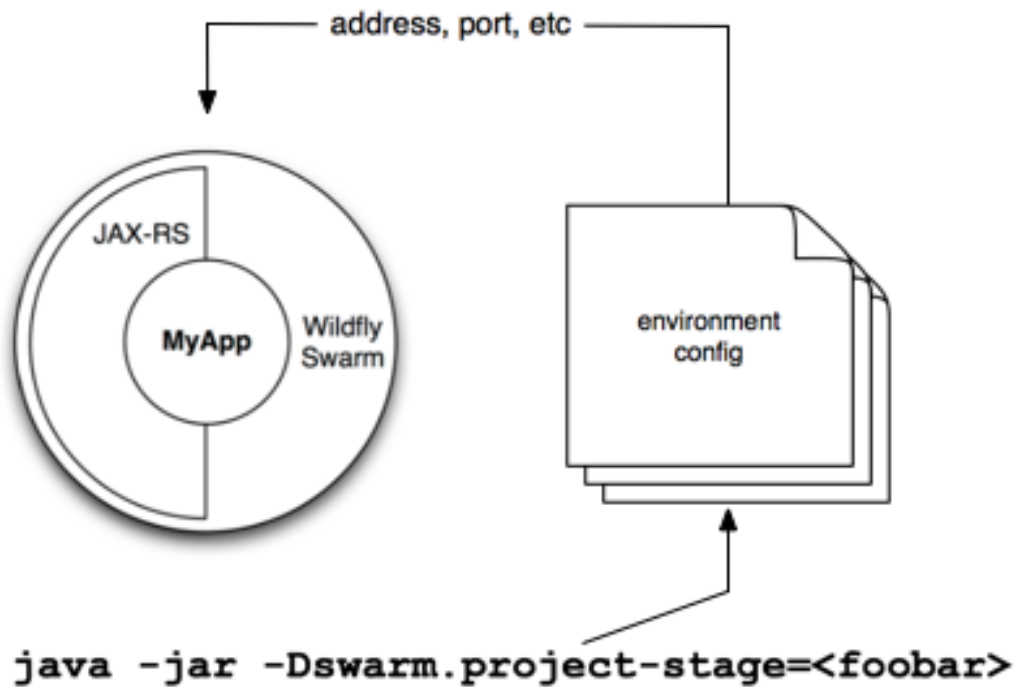
Setup JDBC driver and Datasource

```
public static void main(String[] args) throws Exception {
    Swarm swarm = new Swarm();
    swarm.fraction(
        new DatasourcesFraction()
            .jdbcDriver("h2", (d) -> {
                d.driverClassName("org.h2.Driver");
                d.xaDataSourceClass("org.h2.jdbcx.JdbcDataSource");
                d.driverModuleName("com.h2database.h2");
            })
            .dataSource("ExampleDS", (ds) -> {
                ds.driverName("h2");
                ds.connectionUrl("jdbc:h2:mem:test;DB_CLOSE_DELAY=-1;DB_CLOSE_ON_EXIT=FALSE");
                ds.userName("sa");
                ds.password("sa");
            })
    );
    swarm.start().deploy();
}
```

Environment specific configuration

`./src/main/resources/project-stages.yml`

```
logger:
  level: DEBUG
database:
  jdbc:
    url: foo
---
project:
  stage: development
logger:
  level: DEBUG
database:
  jdbc:
    url: bar
---
project:
  stage: production
logger:
  level: INFO
database:
  jdbc:
    url: somethingElse
```



Resources



- wildfly-swarm.io
- github.com/wildfly-swarm
- [@wildflyswarm](https://twitter.com/wildflyswarm)
- [irc: #wildfly-swarm](https://irc.freenode.net/#wildfly-swarm)



- www.wildfly.org
- github.com/wildfly
- [@wildflyAS](https://twitter.com/wildflyAS)
- wildfly.org/joinus

wildfly-swarm.gitbooks.io/wildfly-swarm-users-guide/