jboss.org

JBoss Reloaded

Carlo de Wolf

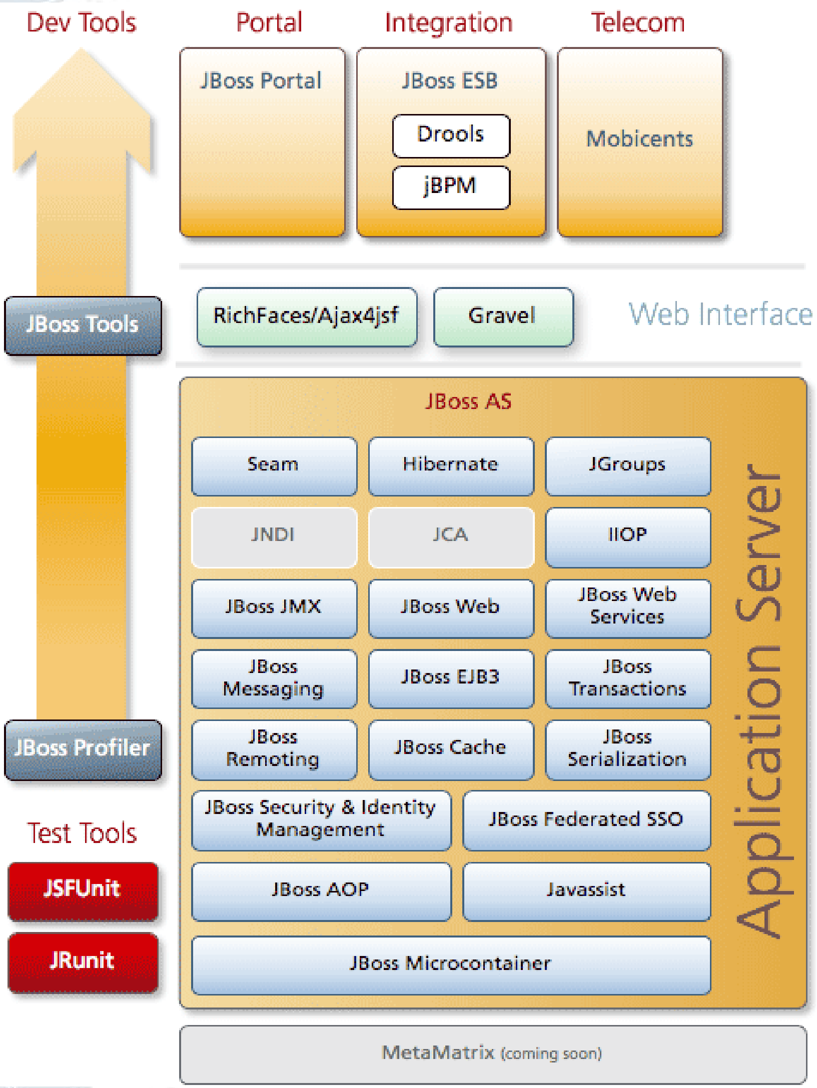
JBoss EJB3 Lead Developer

JBUG Benelux

Agenda

- JBoss Application Server Architecture
- What is Reloaded?
- Why do I need Reloaded?
- JBoss Bootstrap
- A Microcontainer Service
- Starting Reloaded
- Peeking inside
- Recap

JBoss AS Architecture



What is it?

- JBoss MicroContainer provides a means to manage services
- JBoss Deployer provides a deployer framework
- JBoss Reloaded defines the Virtual Deployer Framework using JBoss Bootstrap
- org.jboss.deployers.client.spi.DeployerClient

Why do I need it?

- Finer grained service enablement
- Access to services from within your application

bootstrap.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
   The list of URLs for mc beans to load during bootstrap.
-->
<bootstrap xmlns="urn:jboss:bootstrap:1.0">

   <url>bootstrap/logging.xml</url>
   <url>bootstrap/vfs.xml</url>
   <url>bootstrap/classloader.xml</url>
   <url>bootstrap/aop.xml</url>
   <url>bootstrap/jmx.xml</url>
   <url>bootstrap/deployers.xml</url>
   <url>bootstrap/bindings.xml</url>
   <url>bootstrap/profile.xml</url>

</bootstrap>
```

service.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!--
   My service configuration
-->
<deployment xmlns="urn:jboss:bean-deployer:2.0">

   <classloader>
      <inject bean="myservice-classloader:0.0.0"/>
   </classloader>

   <classloader name="myservice-classloader"
    xmlns="urn:jboss:classloader:1.0" export-all="NON_EMPTY" import-all="true">
      <root>${jboss.lib.url}myservice.jar</root>
   </classloader>

   <bean name="MyService" class="org.jboss.reloaded.test.MyService"/>
</deployment>
```

Starting the engine

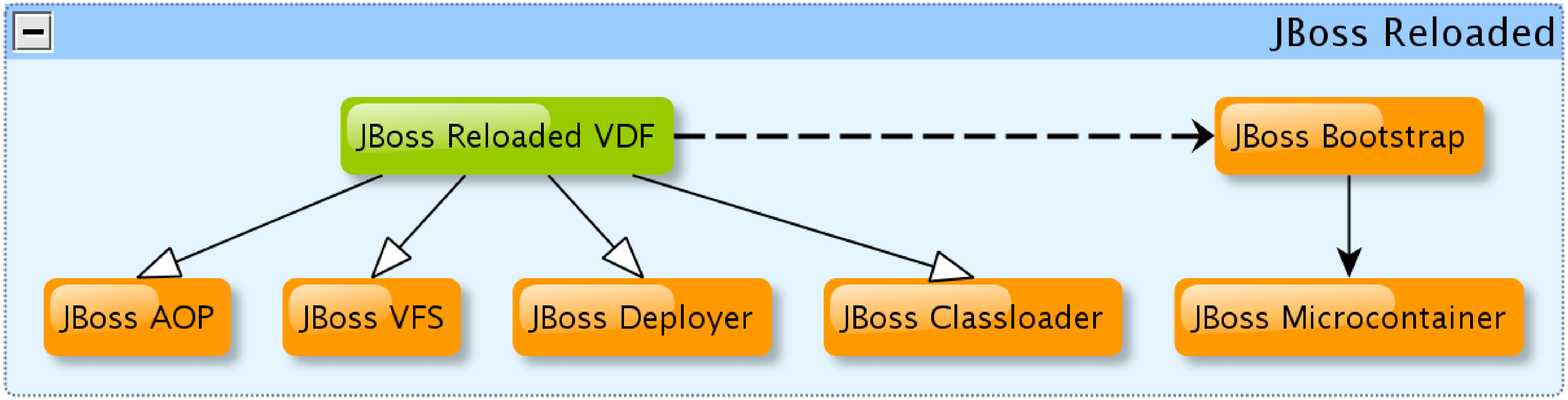
```
VirtualFile roots[] = { };
classLoader =
      ClassLoaderSystem.getInstance().registerClassLoaderPolicy(VFSClassLoaderPolicy.createVFSClassLoaderPolicy(
      "preboot-cl", roots));
URL confURL = classLoader.getResource("conf");
if(confURL == null)
   throw new IllegalStateException("can't find a 'conf' directory on the classpath of " + classLoader);
MCServerConfig config = MCServerConfigFactory.createServerConfig(classLoader);
// To Bootstrap the home is where bootstrap.xml lives
config.bootstrapHome(confURL);
MCServer server = MCServerFactory.createServer(classLoader);
server.setConfiguration(config);
server.initialize();
ClassLoader oldCl = Thread.currentThread().getContextClassLoader();
Thread.currentThread().setContextClassLoader(classLoader);
try
{
   server.start();

   // we're at stage 2
   kernel = server.getKernel();
   mainDeployer = getBean("MainDeployer", ControllerState.INSTALLED, MainDeployer.class);
}
finally
{
   Thread.currentThread().setContextClassLoader(oldCl);
}
```

Peek inside

```
@Test
public void test1()
{
   MainDeployer mainDeployer = (MainDeployer)
   server.getKernel().getController().getInstalledContext("MainDeployer").getTarget();
   Map<String, Deployment> topLevelDeployments = new HashMap<String, Deployment>();
   for(Deployment deployment : mainDeployer.getTopLevel())
      topLevelDeployments.put(deployment.getSimpleName(), deployment);
   VFSDeployment deployment = (VFSDeployment) topLevelDeployments.get("ROOT.war");
   // TODO: will this hold?
   MainDeployerStructure structure = (MainDeployerStructure) mainDeployer;
   // TODO: check to see if we're really running AS 5.0.1
   if(deployment != null)
   {
      DeploymentUnit unit = structure.getDeploymentUnit(deployment.getName());
      // TODO: dive deeper, for now if we can touch the attachments we're good to go
      System.out.println(unit.getAttachments());
   }
   else
      System.err.println("deployment ROOT.war was not found");
}
```

Recap

- JBoss Reloaded Architecture



JBoss Reloaded

JBoss Reloaded VDF

JBoss Bootstrap

JBoss AOP

JBoss VFS

JBoss Deployer

JBoss Classloader

JBoss Microcontainer

Simplifying Startup

```
@BeforeClass
public static void beforeClass() throws DeploymentException, IOException
{
  Properties properties = new Properties();
  container = EJBContainer.createEJBContainer(properties);

  on(container).deploy(deployment(resource("explicit/ds-deployers-beans.xml")));

  on(container).deploy(
      deployment("default-ds",
        NonXADataSourceDeploymentMetaData.class,
        localDataSource()
          .connectionURL("jdbc:hsqldb:mem:defaultdb")
          .driverClass("org.jboss.ejb3.embedded.test.common.HSQLDBService")
          .jndiName("java:/DefaultDS")
          .user("sa")
          .password("")
          .getMetaData()
      ),
      deployment(
          pkg("org.jboss.ejb3.embedded.test.jpa"),
          persistence(
              unit("tempdb")
                .jtaDataSource("java:/DefaultDS")
                .property("hibernate.hbm2ddl.auto", "create-drop")
              ))
      );
}
```

Q&A

- http://www.jboss.org/community/wiki/JBossReloadedDesign
- https://jira.jboss.org/jira/browse/RELOADED
- https://jira.jboss.org/jira/browse/JBBOOT
- http://www.jboss.org/community/wiki/EJB31Embeddable