



Building Java EE 6 applications with JBoss

2012-04-11

Martin Kouba

“Today's goal is to answer...
Is Java EE 6
awesome?”

Max Rydahl Andersen *

* <http://www.slideshare.net/maxandersen/see-context-dependency-injection-from-java-ee-6-in-action>

“Today's goal is to answer...
Is Java EE 6
usable?”

Agenda

1. Programming model → EE 6 technologies
2. Runtime environment → Application Server
3. Quality → Testing platform

Programming model

Java EE 6

What's new in Java EE 6?

- goals: “*ease of development, extensibility, flexibility*”
- notable updates:
 - Servlet 3.0, EJB 3.1, JPA 2, JSF 2
- new technologies:
 - CDI 1.0, Bean Validation 1.0, JAX-RS 1.1
- profiles and pruning
- [Introducing the Java EE 6 Platform](#)

Java EE 6 profiles

- profile = configuration of the Java EE platform targeted at a specific class of applications
- Java EE 6 Web Profile
 - Servlet 3.0, JSP 2.2, EL 1.2, JSF 2.0, EJB 3.1 Lite, JTA 1.1, JPA 2.0, Bean Validation 1.0, CDI 1.0, ...
 - no WS APIs (JAX-RS, JAX-WS, ...)

Java EE 6
to be continued...

Runtime Application Server

JBoss Application Server 7

- Java EE 6 Compatibility *
 - 7.0.x Web Profile
 - 7.1.x Full Platform
- goals: fast start-up, lightweight, modular design, easy configuration, ...

* <http://www.oracle.com/technetwork/java/javaee/overview/compatibility-jsp-136984.html>

JBoss AS 7.1 quickstarts

- [list of available quickstarts](#)

Quality Testing platform

Integration testing à la JBoss

- test (micro)deployments in the real container
- no build, container lifecycle and deployment hassles



ShrinkWrap +



Arquillian

1.0.0-cr-1

1.0.0.CR8

ShrinkWrap

- ***“Skip the Build!”***
- fluent API to assemble archives

```
WebArchive archive =  
    ShrinkWrap.create(WebArchive.class)  
        .addClass(Foo.class)  
        .addResource("my.properties");
```

ShrinkWrap - useful subprojects

- **Descriptors** 1.2.0-alpha-2
 - fluent API to assemble XML descriptors (web.xml, beans.xml, ...)
- **Resolvers – Maven impl.** 2.0.0-alpha-1
 - fluent API for resolving dependencies from a Maven backend

Arquillian's business

- **[start/stop container]**
- **deploy/undeploy** ShrinkWrap archives
- **[enrich the test class]** (@EJB, @Inject, ...)
- **runs the test** inside the active container
- **collect and propagate test results**



Arquillian – other features

- integrates with **JUnit** and **TestNG**
 - however does not allow to use all features
- is **extensible**
 - feel free to write your own extension or container adapter

Arquillian - container adapters

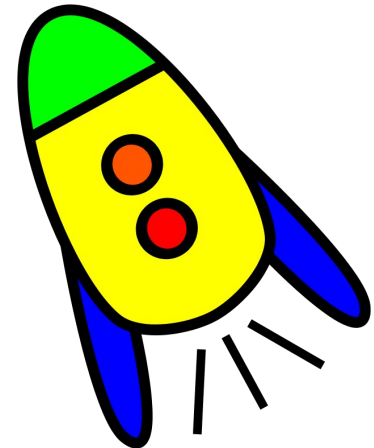
- [list of supported containers](#)
 - JBoss AS 5.1, 6, 7
 - GlassFish 3.1
 - Jetty 6, 7
 - Tomcat 6, 7
 - WAS
 - WLS
 - ...

Arquillian - useful links

- Info & Guides
 - <http://arquillian.org/>
 - <http://arquillian.org/guides/>
- Showcase
 - <https://github.com/arquillian/arquillian-showcase>
- Functional testing
 - <https://community.jboss.org/en/arquillian/blog/2011/09/17/functional-testing-with-arquillian>

Arquillian - time for simple example

- testing EJB 3.1 components
- test enrichment
- debugging
- inspecting deployment archive



Java EE 6

back to the future...

EE 6 update - Servlet 3.0

- configuration
 - new annotations `@WebServlet`, `@WebFilter`, ...
 - `WEB-INF/classes`, `WEB-INF/lib`
 - `web.xml` is optional; but overrides annotations
 - `web.xml` fragments (modularization)
- asynchronous processing

EE 6 update - EJB 3.1

- no-interface view
- singleton session bean
- asynchronous session bean invocations
- simplified packaging (deploy EJB directly in a web archive)
- portable global JNDI name syntax
- timer service improvements

EE 6 update - JPA 2

- mapping enhancements
- standard properties
- pessimistic locking
- API additions
(detach, query hints, cache, ...)
- Criteria API
- JPQL enhancements

EE 6 update - JSF 2

- <http://andyschwartz.wordpress.com/2009/07/31/whats-new-in-jsf-2/>
- facelets as default view declaration language
- ajax integration
- state saving improvements
- implicit and conditional navigation
- bean validation integration
- resource loading
- ...

EE 6 newcomer - CDI 1.0

- CDI = Contexts and Dependency Injection
 - a well-defined lifecycle for stateful objects bound to lifecycle contexts (the set of contexts is extensible)
 - typesafe dependency injection
 - events, interceptors, decorators, ...
- reference implementation → **Weld**



EE 6 newcomer - JAX-RS 1.1

- “... for the development of web services built according to the *REST* architectural style.”
- Java API for RESTful Web Services
- JBoss implementation → **RESTEasy**



EE 6 newcomer - Bean Validation 1.0

- “... defines a metadata model and API for *JavaBean validation...*”
 - the default metadata source = annotations
 - not tied to any app tier (e.g. web or persistence)
- reference implementation → **Hibernate Validator**



That's all...

Thanks for listening