JAX-RS 3.0

The Next Generation

Markus KARG
markus@headcrashing.eu
About Me
Application Servers: Old-School Scaling

Clustered Application Server Nodes
e.g. GlassFish including Jersey

Java EE Application Server
- JAX-RS Application
- JAX-RS Container
- Java Runtime
- Operating System
- Bare Metal

AS-specific Cluster
- AS-spec. Cluster Controller
- AS-spec. Node 1
- AS-spec. Node 2
- AS-spec. Node ...
- AS-spec. Node n

AS-specific Cluster
Cloud Computing: The Modern Approach

Elastic creation and disposal of lightweight processes (e.g. Kubernetes with Docker)
Cloud Computing

JAX-RS + Netty + Docker + ... = Cloud Native Java
RESTful Devices

5 x 2,5 cm
1 Core @ 1 GHz
256 MB RAM
100 Mb/s LAN
$ 10

Try to start GlassFish...
Effects upon JAX-RS

- should start and stop instantly
- should consume less resources
- must provide http server
- must read external config
- must accept application-provided resources
Java SE 11: JAXB → <dependency>

JAX-RS 2.2: javax.ws.rs → jakarta.ws.rs

JAX-RS 3.0: @Context → @Inject
Roadmap

JAX-RS 2.2 [Q1/2019]: Java SE Bootstrap API
(instant on/off, low resource, includes http server, reads external config incl. optional support for Microprofile Config API)

JAX-RS 2.3 [Q2/2019]: CDI 2.0 Support – Deprecate @Context
(accept application-provided resources)

https://github.com/eclipse-ee4j/jaxrs-api/wiki/Roadmap
Further Roadmap

JAX-RS 2.4 [TBD]: JPMS, Java 9

JAX-RS 3.0 [Q4/2019]: Replace `@Context` by `@Inject`

**WARNING:** You MUST use CDI now!

JAX-RS 3.1 [TBD]: Flow API, NIO

...
JAX-RS 2.2: Java SE Bootstrap API

- instant on/off – boot in one second
- low resource – run on limited devices
- includes http server – choose by actual need
- reads external config – from app, controller or orchestrator
- optional support for Microprofile Config API – cloud standards
Bleeding Edge Cookbook

Java SE 8, Maven 3.6.1

maven-shade-plugin: master

jax-rs 2.1: d5f18229e3d57ffbb8960ca2b19b51893c946be0a

jersey: JAXRS_2_2
JAX-RS 2.2: Java SE Bootstrap API

(live coding)
<table>
<thead>
<tr>
<th>Platform (1 CPU Core)</th>
<th>Jersey + Netty JAX-RS 2.2</th>
<th>Microprofile JAX-RS 2.1</th>
<th>Java EE Server JAX-RS 2.1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Boot Time</strong></td>
<td>~ 2.5s</td>
<td>(~ 10s)</td>
<td>(~ 60s)</td>
</tr>
<tr>
<td><strong>JAR Size</strong></td>
<td>6 MB</td>
<td>70 MB</td>
<td>142 MB</td>
</tr>
<tr>
<td><strong>Java -Xmx</strong></td>
<td>6 MB</td>
<td>(?)</td>
<td>(?)</td>
</tr>
<tr>
<td><strong>Java -Xms</strong></td>
<td>1 MB</td>
<td>(?)</td>
<td>(?)</td>
</tr>
<tr>
<td><strong>Docker -m</strong></td>
<td>&lt; 32 MB</td>
<td>(?)</td>
<td>(?)</td>
</tr>
</tbody>
</table>
Questions?

https://github.com/eclipse-ee4j/jaxrs-api

Markus KARG (JUG Goldstadt)
JAX-RS Committer
http://www.headcrashing.eu
markus@headcrashing.eu