

Developing RESTful Web Services With JAX-RS and Jersey

Jakub.Podlesak@Sun.COM

<http://blogs.sun.com/japod>

SUN Microsystems

Goal

To show you how to develop RESTful web services using Java

REpresentational State Transfer

REST is an architectural style

Set of constraints

Applied to the architecture of a distributed system

Key Concepts

Resources identified by IDs

Uniform service interface

Hypermedia exchange

Stateless communication

Why apply REST?

It works :-)

Simplicity

Loose coupling

Scalability

Service Composition & Orchestration

JAX-RS (1/2)

Standardized in the JCP: JSR-311
Server side centric
Part of Java EE 6

JAX-RS (2/2)

POJO based

Annotation driven

Container independent

JAX-RS Example (1/2)

URI mapping

@Path("/user/{id}")

```
public class UserResource {
```

HTTP method

@GET @Produces("application/xml")

```
public User getUser(@PathParam("id") String id) {
```

```
    return lookupUser(id);
```

```
}
```

```
}
```

Media type

Parameter mapping

JAX-RS Example (2/2)

@Path("/user/{id}")

```
public class UserResource {
```

@PUT @Consumes("application/xml")

```
public void putUser(@PathParam("id") String id,
```

```
                    User u) {
```

```
    return updateUser(id, u);
```

```
}
```

```
}
```

Body entity mapping

Demonstration

Matching URIs (@Path, @PathParam)

Processing GET/PUT requests

Producing/consuming with JAXB

JSON with JAXB

Content negotiation

Processing query params (@QueryParam)

Building responses with Response builder

Jersey

Open source

Production quality

Reference implementation of JAX-RS

1.1.0-ea released on Monday

Jersey features

Jersey client API

Embedded Grizzly container

Embedded GlassFish

JSON with JAXB

MVC with JSP

Multipart MIME

Simple injection

Jersey features (contd...)

Spring integration

Guice integration

Java client API using Apache HTTP client

Comet support

Quick start

Maven archetypes

```
mvn -DarchetypeCatalog=
```

```
http://download.java.net/maven/2/archetype-catalog.xml
```

```
archetype:generate
```

GlassFish

3.0 Preview (Jersey core libs integrated directly)

2.1 (Jersey libs and apidocs and examples at UC repo)

NetBeans 6.5

GET more information

<https://jsr311.dev.java.net/>

<https://jersey.dev.java.net/>

Blogs

<http://blogs.sun.com/sandoz>

<http://weblogs.java.net/blog/mhadley/>

<http://blogs.sun.com/japod>

<http://blogs.sun.com/theaquarium/>

Q&A

<http://blogs.sun.com/japod>

Thank You!

And do not forget to visit:

<https://jersey.dev.java.net>