The Java EE 6 Platform

Alexis Moussine-Pouchkine
GlassFish Team

This is no science fiction

Java EE 6 and GlassFish v3 shipped **final releases** on December 10th 2009

A brief History

Project JPE

May 1998

Enterprise Application

J2EE 1.2

Servlet
JSP
EJB
JMS
RMI/IIOP

Dec 1999
10 specs

Robust Scalable

J2EE 1.3

CMP
JCA

Sept 2001
13 specs

Web Services

J2EE 1.4

WS
Management
Deployment

Nov 2003
20 specs

Ease of development

Java EE 5

Annotations
EJB 3
JPA 1.0
WS-*
JSF

May 2006
23 specs

Ease of development (web)

Java EE 6

EJB 3.1
JPA 2.0
Servlet 3.0
JSF 2.0
JAX-RS 1.1
JCDI 1.0
@Inject
Bean Validat°

Web Profile

Managed Bean

Q4 2009
28 specs

Java EE 6 – What's New?

- Several new APIs
- Web Profile
- Extensibility & Pluggability
- Dependency Injection
- Improvement to many APIs

New and improved specifications

- EJB 3.1
- JPA 2.0
- Servlet 3.0
- JSF 2.0
- JAX-RS 1.1
- Connectors 1.6
- Bean Validation 1.0
- DI 1.0
- CDI 1.0
- Managed Beans 1.0
- Interceptors 1.1
- JAX-WS 2.2
- JSR-109 1.3
- JSP 2.2 / EL 2.2
- JSR-250 1.1

JAX-RS

- RESTful web services API
- Already widely adopted
- Really a general, high-level HTTP API
- Annotation-based programming model
- Programmatic API when needed
- JAX-RS 1.1 integration with EJBs

JAX-RS sample code

```
@Path("widgets/{id}")
@Produces("application/widgets+xml")
public class WidgetResource {
    public WidgetResource(
            @PathParam("id") String id) {
        ...
    }

    @GET
    Widget getWidget() {
        ...
    }
}
```

Bean Validation 1.0

```
public class Address {
    @NotNull @Size(max=30,
        message="longer than {max} characters")
    private String street1;
    ...
    @NotNull @Valid
    private Country country;
}

public class Country {
    @NotNull @Size(max=20)
    private String name;
    ...
}
```

request recursive object graph validation

Build your own!

```
@Size(min=5, max=5)
@ConstraintValidator(ZipcodeValidator.class)
@Documented
@Target({ANNOTATION_TYPE, METHOD, FIELD})
@Retention(RUNTIME)
public @interface ZipCode {
    String message() default "Wrong zipcode";
    String[] groups() default {};
}
```

- Integrated in JPA and JSF
- Bootstrap APIs

“Web” Profile

- First Java EE profile to be defined
- A fully-functional, mid-size stack for modern web application development
- Complete, but not the kitchen sink

Java EE 6 Web Profile

- Servlet 3.0
- JSP 2.2 / EL 2.2
- JSR-45 1.0
- JSTL 1.2
- JSF 2.0
- Bean Validation 1.0
- EJB 3.1 Lite
- JPA 2.0
- JTA 1.1
- DI 1.0
- CDI 1.0
- Managed Beans 1.0
- Interceptors 1.1
- JSR-250 1.1

Extensible & Pluggable

- Focus on the web tier in this release
- Create a level playing ground for third-party libraries and frameworks
- Simplify packaging of web apps

Modular Web Applications

- Libraries can contain `web-fragment.xml`
- `web.xml` is optional
- `@WebServlet`, `@WebFilter` annotations
- `ServletContainerInitializer` interface
- Programmatic registration
- Resource jars

JSF 2.0

- Standardized facelets
- Auto-discovery of component libraries
- Composite components
- Ajax support with partial views
- Even a JavaScript API !

EJB 3.1

- `@Singleton` beans
- `@Startup` beans
- `@Asynchronous` invocations
- `@Schedule` tasks
- Define EJBs directly inside a web app
- `EJBContainer` API works on Java SE

Dependency Injection

- JSR-299 + JSR-330
- `@Resource` still around

```
@Resource DataSource myDB;
```

- Added `@Inject` support

```
@Inject @LoggedIn User user;
```

- Beans discovered at startup
- Injection metamodel (`BeanManager` API)

Well-defined Conversation Scope

- A *conversation* is :
  - explicitly demarcated
  - associated with individual browser tabs
  - accessible from any JSF request

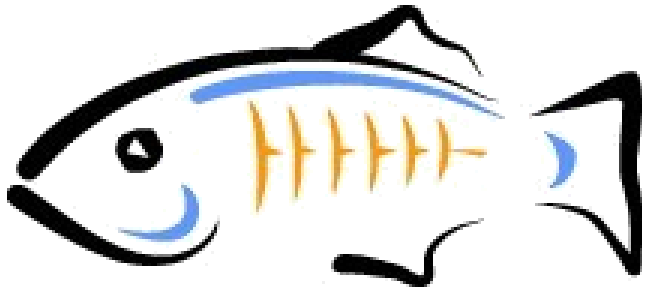
```
@Named
@ConversationScoped
public class ItemFacade implements Serializable {
  @Inject Conversation conversation;
  ...
  conversation.begin(); // long-running
  ...
  conversation.end();   // schedule for destruction
```

Many ways to simplify things

- Capture common patterns
- Fix inconsistencies
- Adopt what works
- Make APIs work better together
- Reduce packaging overhead
- Be transparent

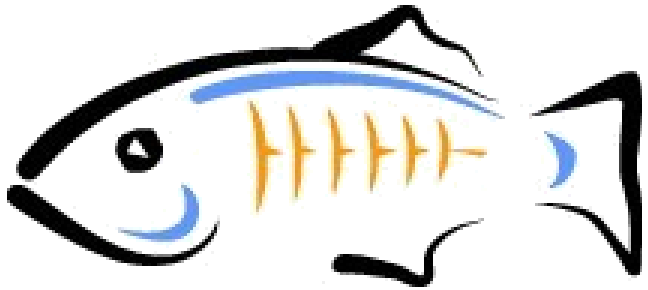
GlassFish v3 – for the developer

- Java EE 6
- Development tools integration
- Deploy on save
- Preserve session across redeploys
- Update Center

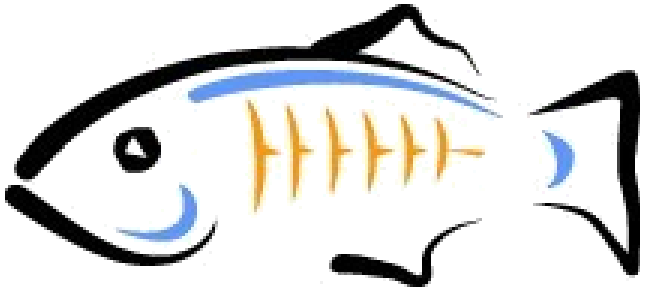


GlassFish v3 – Modularity

- Based on industry standard OSGi
- Pay for what you use architecture
- OSGi bundles deployable straight in GlassFish v3
- `@Resource` to inject an OSGi service



DEMO



Thanks for your attention!

- http://java.sun.com/javaee
- http://jcp.org/en/jsr/summary?id=316
- Java EE 6 and GlassFish v3 Virtual Conference http://www.sun.com/events/javaee6glassfishv3/virtualconference/index.jsp
- *"Introducing the Java EE 6 Platform"* article http://java.sun.com/developer/technicalArticles/JavaEE/JavaEE6Overview.html
- http://glassfish.org
- http://beginningee6.kenai.com/ (tutorial code)
- http://javaee-patterns.kenai.com/ (Adam Bien)

alexis.mp@sun.com
http://blogs.sun.com/alexismp
twitter:alexismp