

Java Platform, Enterprise Edition (Java EE) 6 Overview

Roberto Chinnici
Sun Microsystems, Inc.
2940

JAZOON08

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY
JUNE 23 - 26, 2008 ZÜRICH



Agenda

- > Java EE Platform evolution
- > Theme and features
- > Components (JSRs)
- > Timeline
- > Get involved!
- > The Great Web Profile Poll

Java EE 5



JavaTM
COMPATIBLE
ENTERPRISE EDITION

Ease of Development Focus

Major Revamp of Programming Model

EJBTM 3.0 support for POJOs means less to learn, less to code and less to maintain

New JavaTM Persistence API makes object/relational mapping cleaner and easier

New and updated Web Services (JAX-WS 2.0 and JAXB 2.0) simplifies SOA implementation

JavaServerTM Faces 1.2 facilitates building Web 2.0 Applications with AJAX

Annotations often eliminate the need for deployment descriptors

Supported by NetBeansTM 5.5 Enterprise Pack

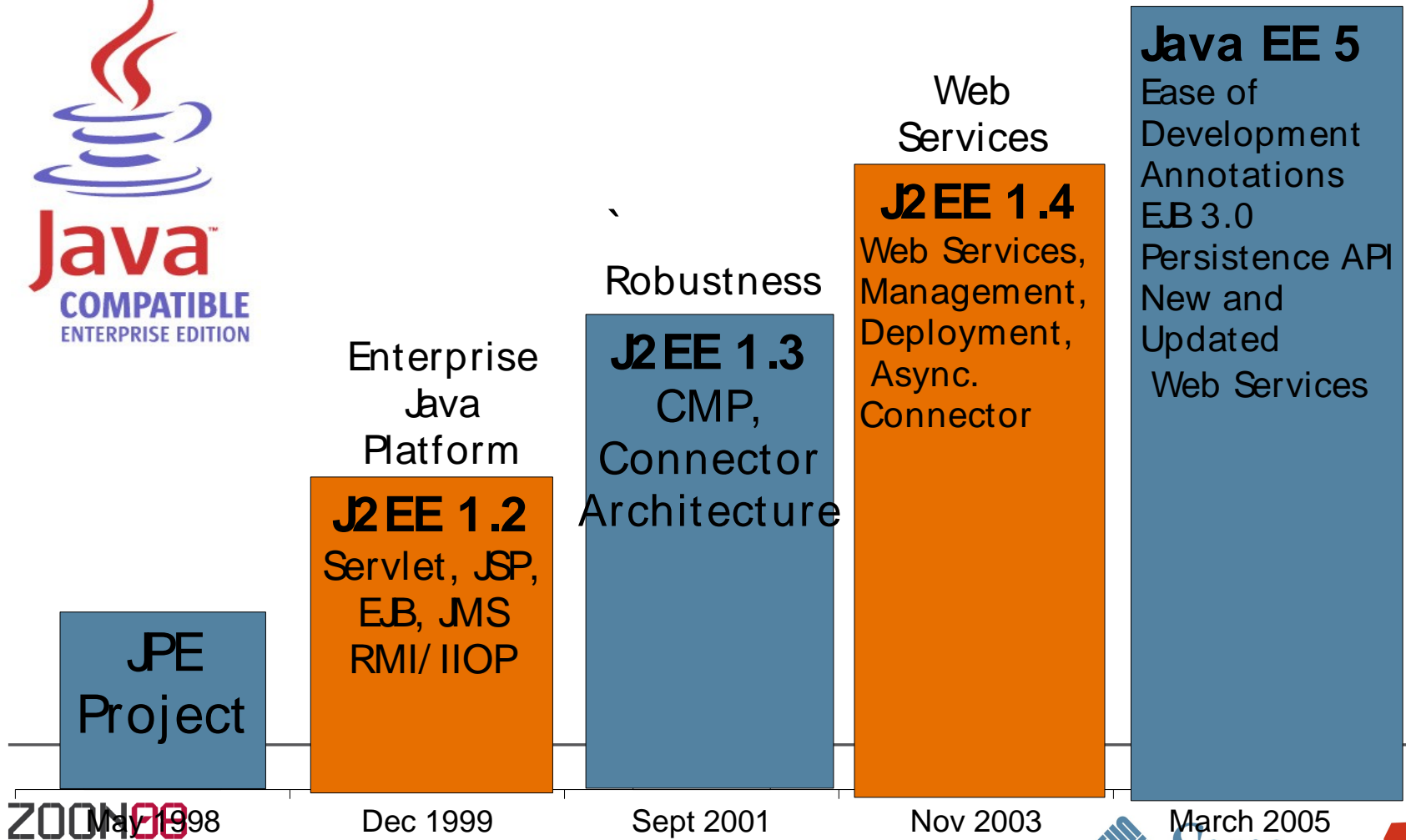
Get the SDK :

<http://java.sun.com/javae/downloads/>

Java EE 5 Implementations



History of Enterprise Java



Theme for Java EE 6

Rightsizing

=

Flexibility, Focus, Simplification

Rightsizing Explained

- > Enable **flexibility** at development and deployment
- > Make it possible to **focus** on specific classes of applications
- > **Simplify** by removing rarely used or superseded APIs

New Java EE 6 Features

- > Profiles
- > Pruning
- > Extensibility
- > More ease of development

Profiles

- > Rules defined by the Java EE Platform spec
- > Profiles can be
 - proper subsets
 - proper supersets
 - overlapping (drop something, add something else)
- > Web Profile defined right away
- > Other profiles follow regular JCP process

Pruning

- > Make some components optional
- > Strong signal for all new development work
- > Same rules proposed by Java SE
 - “pruned now, optional in next release”
- > Pruning candidates:
 - JAX- RPC, EJB Entity Beans
 - JAXR, JSR- 88

Extensibility

- > Embrace open source libraries and frameworks
- > Create a level playing field
- > Enable simple pluggability of technologies
- > Scripting as a first-class citizen

Ease of Development in the Web Tier

- > Use of annotations across all web APIs
- > No editing of web.xml required
- > Self- registration of third- party libraries
- > Simplified packaging
- > JAX- RS for RESTful web services
- > Scripting support via extensibility

JAX- RS Sample

```
import javax.ws.rs.*;
import org.apache.abdera.model.*;
@Path("activitylog")
@ProducesMime("application/atom+xml")
public class ActivityLog {
    @GET
    public Feed getFeed() {...}

    @GET @Path("{id}")
    public Entry getEntry(@PathParam("id") String id) {...}
}
```

EJB 3.1

- > Singletons
- > Optional local interfaces
- > Simplified packaging in web apps
- > Application startup/ shutdown events
- > Calendar- based timers
- > Global JNDI names
- > EJB “Lite”

EJB 3.1 Sample (1)

@Startup

@Singleton

```
public class SingleBean {
```

```
    @PostConstruct
```

```
    public void onStartUp() {
```

```
        ... application initialization code goes here ...
```

```
    }
```

```
    ... more methods to access shared data ...
```

```
}
```

EJB 3.1 Sample (2)

`@Stateless`

```
public class TimerBean {
```

```
    @Schedule(dayOfWeek= "Sun")
```

```
    public void onSunday() {
```

```
        ... periodically do some work ...
```

```
    }
```

```
}
```

EJB 3.1 Sample (3)

@Stateless

```
public class HelloWorldBean /* no interface */ {  
    public String hello(String name) {  
        return "hello, " + name;  
    }  
}
```

@EJB HelloWorldBean hello;

hello.hello("David");

New JSRs in Java EE 6

- > EJB 3.1
- > JPA 2.0
- > Servlet 3.0
- > JSF 2.1
- > JAX- RS 1.0
- > WebBeans 1.0
- > Java Connector Architecture 1.6

More JSRs We Track

- > Bean Validation (JSR- 303)
- > Timer API (JSR- 236)
- > Work Manager API (JSR- 237)

Other Components

- > Java SE 6 underneath
- > Point updates:
 - JAX-WS 2.2
 - JSR-109 1.3
 - JSP 2.2
- > Completed JSRs:
 - JSR-196
 - JAXB 2.2

Timeline

- > June 2007 Expert group formed
- > Q3 2008 Public draft
- > Q4 2008 Proposed final draft
- > Q2 2009 Final release

Participate!

- > Early/ public drafts available for many of the specs
- > Send feedback
- > Contribute to the GlassFish implementation
- > Use TP2, blog about it

GlassFish V3

- > Open source implementation of Java EE 6
- > Fully modular, extensible
- > OSGi R4 based
- > Small footprint, fast startup
- > Multiple containers, including JRuby, Groovy

Web Profile Options (from my blog)

> A

servlets, JSP, JSTL, EL, JSR- 45, JSR- 250

> B = A +

EJB Lite, JTA, JPA, JSF, WebBeans

> C = Full Platform = B +

JAX- RS, Connectors, JMS, JAF, JavaMail,

JAX- WS, JAXB, JSR- 181, JSR- 109,

JSR- 115, JSR- 196, JSR- 77,

JAX- RPC*, JAXR*, JSR- 88*

Java.net Poll

- > 1. A small set of JSRs
- > 2. A larger set of JSRs without JSR or WebBeans
- > 3. A larger set of JSRs with JSR and WebBeans
- > 4. I'm not interested in Java EE 6 profiles

Java.net Poll (Results)

- > 1. A small set of JSRs
17%
- > 2. A larger set of JSRs without JSR or
11.9%
WebBeans
- > 3. A larger set of JSRs with JSR and WebBeans
46.7%
- > 4. I'm not interested in Java EE 6 profiles
24.2%

Q&A

JAZOON08

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY
JUNE 23 - 26, 2008 ZÜRICH



Roberto Chinnici

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

Sun Microsystems, Inc. roberto.chinnici@sun.com

JAZOON08

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY
JUNE 23 - 26, 2008 ZURICH

