

Java EE 6 & GlassFish v3

Paving the path for future

11, 12 Dec. 2009

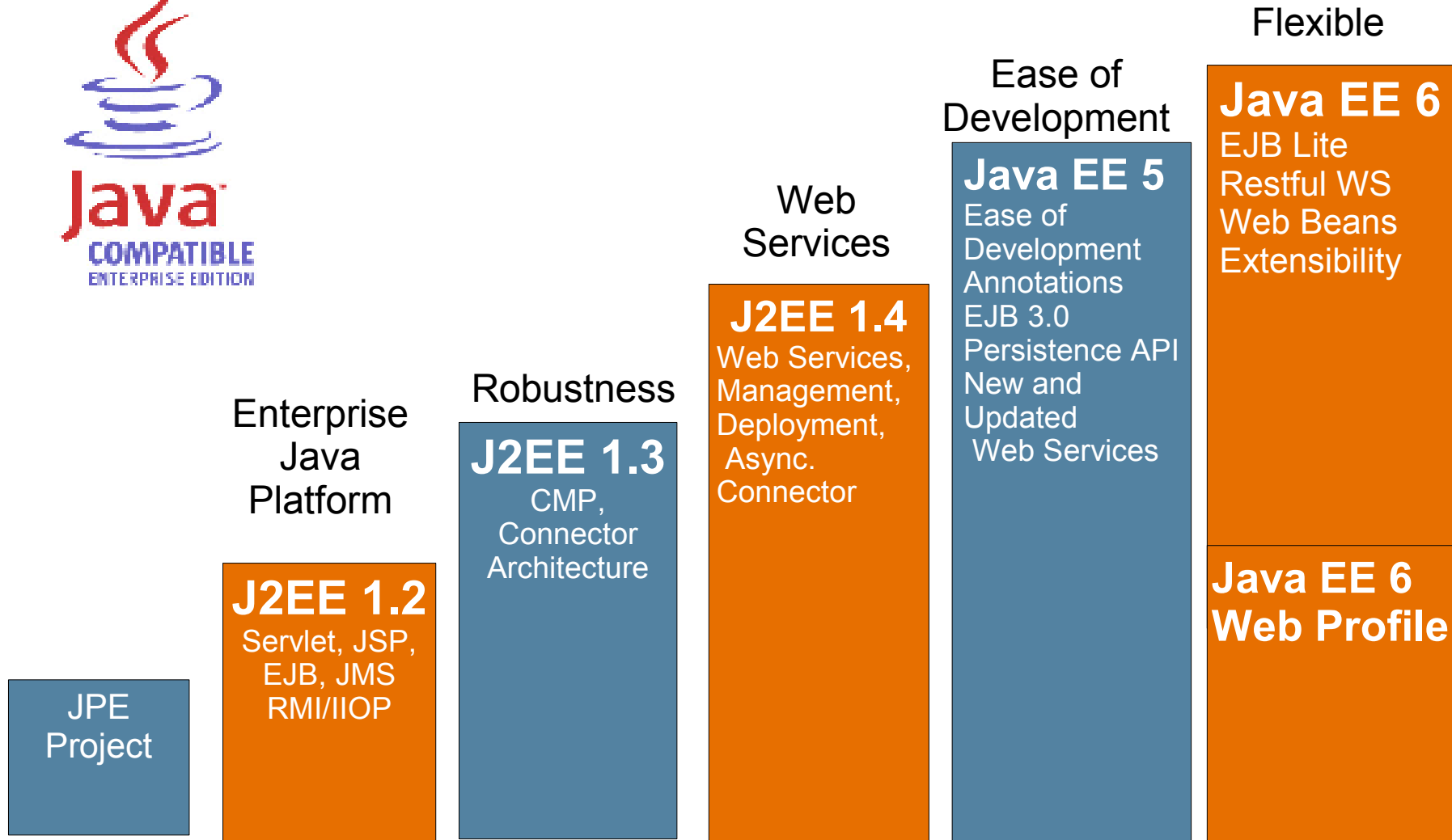
Pune, India

IndicThreads.com Conference On Java Technology

Arun Gupta (blogs.sun.com/arungupta, @arungupta)

Sun Microsystems, Inc.

Java EE: Past & Present



Compatible Java EE 5 Implementations



Goals for the Java EE 6 Platform

- Flexible & Light-weight
- Extensible
 - Embrace Open Source Frameworks
- Easier to use, develop on
 - Continue on path set by Java EE 5



Java EE 6 is Flexible

- Decouple specifications to allow more combinations
- Expands potential licensee ecosystem
- Profiles
 - Targeted bundle of technologies
 - Defined through the JCP
 - Web Profile Already Defined
 - Defined by the Java EE 6 Expert Group



Java EE 6 Web Profile 1.0

- Fully functional mid-sized profile
 - Actively discussed in the Java EE 6 Expert Group and outside it
 - Technologies
 - Servlets 3.0, JSP 2.2, EL 2.2, Debugging Support for Other Languages 1.0, JSTL 1.2, JSF 2.0, Common Annotations 1.1, EJB 3.1 Lite, JTA 1.1, JPA 2.0, Bean Validation 1.0, Managed Beans 1.0, Interceptors 1.1, Context & Dependency Injection 1.0, Dependency Injection for Java 1.0



Java EE 6 is Lightweight

- Pruning
 - Make some technologies optional
- Pruned today, means
 - Optional in the next release
 - Deleted in the subsequent releases
- Pruned technologies will be marked in Javadocs
- Technologies



Java EE 6 is Extensible

- Embrace open source frameworks
 - Wicket, Lift, Spring, Struts, ...
- Zero-configuration, drag-and-drop for web frameworks
 - Servlets, servlet filters, context listeners for a framework get discovered and registered automatically
- Plugin libraries using *web fragments*





```
<web-fragment>
  <filter>
    <filter-name>wicket.helloworld</filter-name>
    <filter-class>org.apache.wicket.protocol.http.WicketFilter</filter-class>
    <init-param>
      <param-name>applicationClassName</param-name>
      <param-value>...</param-value>
    </init-param>
  </filter>

  <filter-mapping>
    <filter-name>wicket.helloworld</filter-name>
    <url-pattern>/*</url-pattern>
  </filter-mapping>
</web-fragment>
```

http://blogs.sun.com/arungupta/entry/totd_91_applying_java_ee





```
<web-fragment>
  <filter>
    <filter-name>LiftFilter</filter-name>
    <display-name>Lift Filter</display-name>
    <description>The Filter that intercepts lift calls</description>
    <filter-class>net.liftweb.http.LiftFilter</filter-class>
  </filter>

  <filter-mapping>
    <filter-name>LiftFilter</filter-name>
    <url-pattern>/*</url-pattern>
  </filter-mapping>
</web-fragment>
```

http://blogs.sun.com/arungupta/entry/totd_101_applying_servlet_3



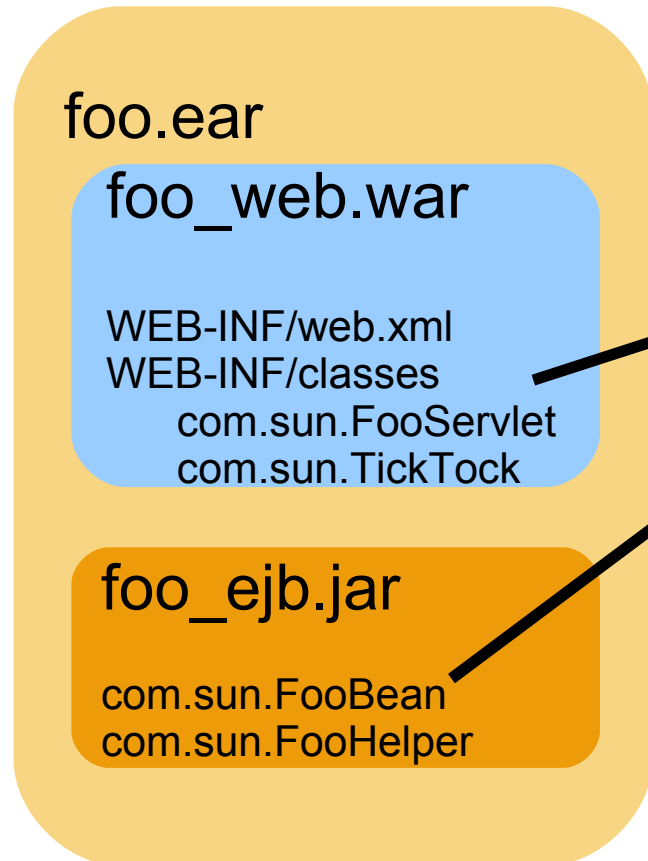
Java EE 6 & Ease-of-development

- Continue advancements of Java EE 5
- Primary focus: Web Tier
- General principles
 - Annotation-based programming model
 - Reduce or eliminate need for DD
 - Traditional API for advanced users

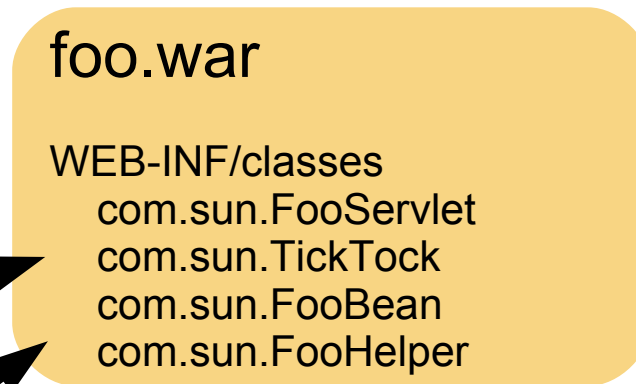


EoD: EJB Simplified Packaging

Java EE 5



Java EE 6



~~web.xml~~ ?

http://blogs.sun.com/arungupta/entry/totd_95_ejb_3_1

EoD: Servlets

Servlet in Java EE 5: Two Files

```
<!--Deployment descriptor
web.xml -->

<web-app>
  <servlet>
    <servlet-name>MyServlet

      </servlet-name>
    <servlet-class>
      com.sun.MyServlet
    </servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>MyServlet
      </servlet-name>
    <url-pattern>/myApp/*
      </url-pattern>
  </servlet-mapping>
  ...
</web-app>
```

```
/* Code in Java Class */

package com.sun;
public class MyServlet extends
HttpServlet {
public void
doGet(HttpServletRequest
req,HttpServletResponse res)
{
...
}
...
}
```



EoD: Servlets in Java EE 6

```
package com.sun;  
@WebServlet(name="MyServlet", urlPattern="/myApp/*")  
public class MyServlet extends HttpServlet {  
    public void doGet(HttpServletRequest req,  
                      HttpServletResponse res)  
    {  
        ...  
    }  
}
```

http://blogs.sun.com/arungupta/entry/totd_81_getting_started_with

Java EE 6 - Done

- Specifications approved by the JCP (Dec 1)
- Reference Implementation is GlassFish v3 (Dec 10)
- TCK (Dec 10)



Java EE 6 Specifications

- The Platform
- Java EE 6 Web Profile 1.0
- Managed Beans 1.0



Java EE 6 Specifications

New

- Context and Dependency Injection for Java EE (JSR 299)
- Bean Validation (JSR 303)
- Java API for RESTful Web Services (JSR 311)
- Dependency Injection for Java (JSR 330)



Java EE 6 Specifications

Extreme Makeover

- Java Server Faces 2.0 (JSR 314)
- Java Servlets 3.0 (JSR 315)
- Java Persistence 2.0 (JSR 317)
- Enterprise Java Beans 3.1 & Interceptors 1.1 (JSR 318)
- Java EE Connector Architecture 1.6 (JSR 322)



Java EE 6 Specifications Updates

- Java API for XML-based Web Services 2.2 (JSR 224)
- Java API for XML Binding 2.2 (JSR 222)
- Web Services Metadata MR3 (JSR 181)
- JSP 2.2/EL 2.2 (JSR 245)
- Web Services for Java EE 1.3 (JSR 109)
- Common Annotations 1.1 (JSR 250)
- Java Authorization Contract for Containers 1.3 (JSR 115)
- Java Authentication Service Provider Interface for Containers 1.0 (JSR 196)



Java EE 6 Specifications

As is

- JDBC 3.0 API
- Java Naming and Directory Interface 1.2
- Java Message Service 1.1
- Java Transaction API 1.1
- Java Transaction Service 1.0
- JavaMail API Specification 1.4
- JavaBeans Activation Framework 1.1
- Java API for XML Processing 1.3
- Java API for XML-based RPC 1.1
- SOAP with Attachments API for Java 1.3
- Java API for XML Registries 1.0
- Java EE Management Specification 1.1 (JSR 77)
- Java EE Deployment Specification 1.2 (JSR 88)
- Java Management Extensions 1.2
- Java Authentication and Authorization Service 1.0
- Debugging Support for Other Languages (JSR 45)
- Standard Tag Library for JSP 1.2 (JSR 52)
- Streaming API for XML 1.0 (JSR 173)



IDE Support

- NetBeans 6.8
- Eclipse 3.4+
- IntelliJ 9.0



Future Directions

- JNLP-ize Java EE app client
- Java EE SPI
 - - Allow service providers and other system level components to be plugged in



Java EE 6 Training & Certification

- Java EE6 Training Curriculum - Feb 2010
- Java EE6 Certifications available - Mar 2010
- Register your interest in the courses and certifications and receive information about promotions
 - https://dct.sun.com/dct/forms/reg_us_1611_480_0.jsp

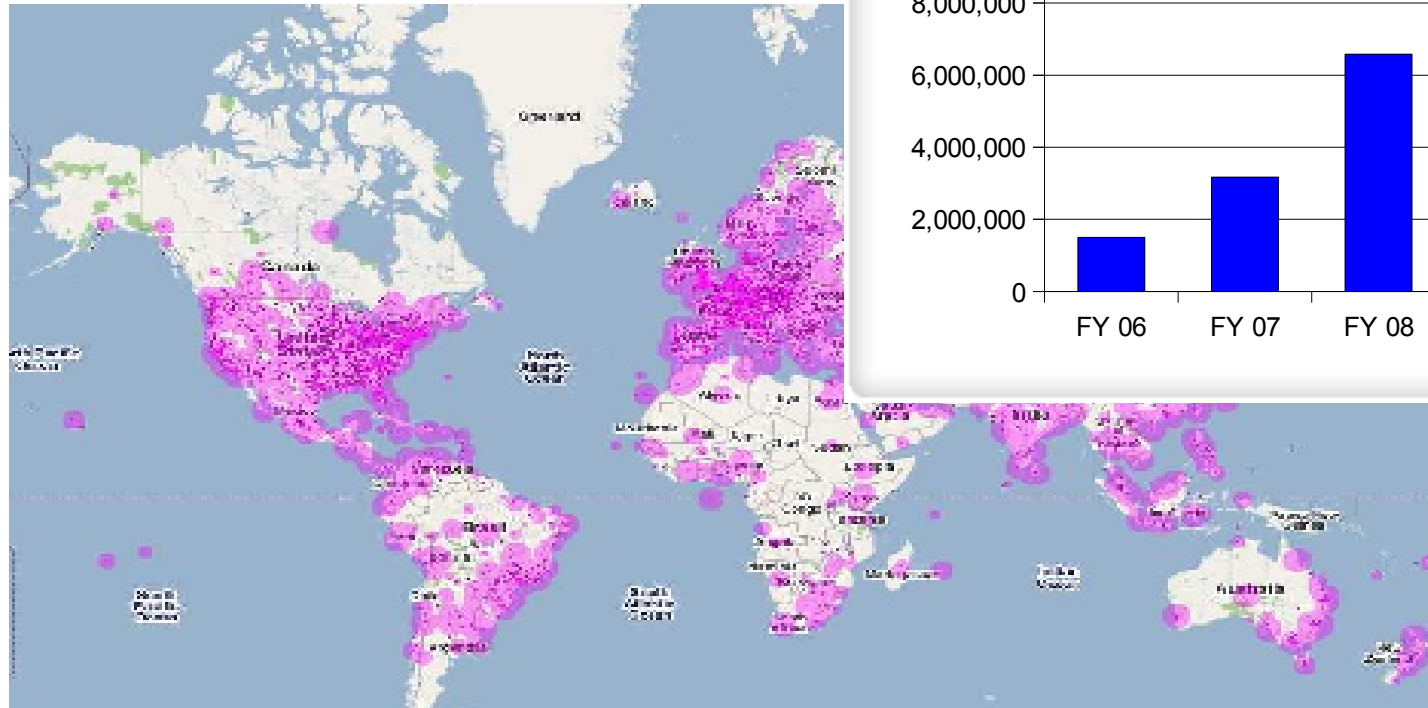


Java EE Adoption Metrics

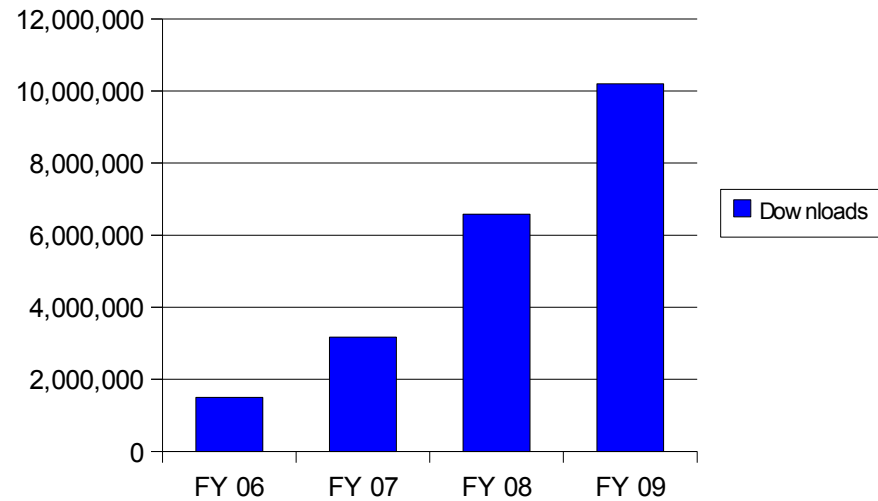
- Over 10 million downloads of Java EE SDKs annually
- Amazon search for J2EE yields 4,167 books (367)
- Amazon search for Java EE 1,681 books (448)
- Java the world's most used programming language
- Monster lists over 2,300 available Java EE / J2EE jobs
- In the past four years:
 - JBoss purchased by Red Hat for 500 million
 - BEA purchased by Oracle for 7 Billion

Java EE Adoption

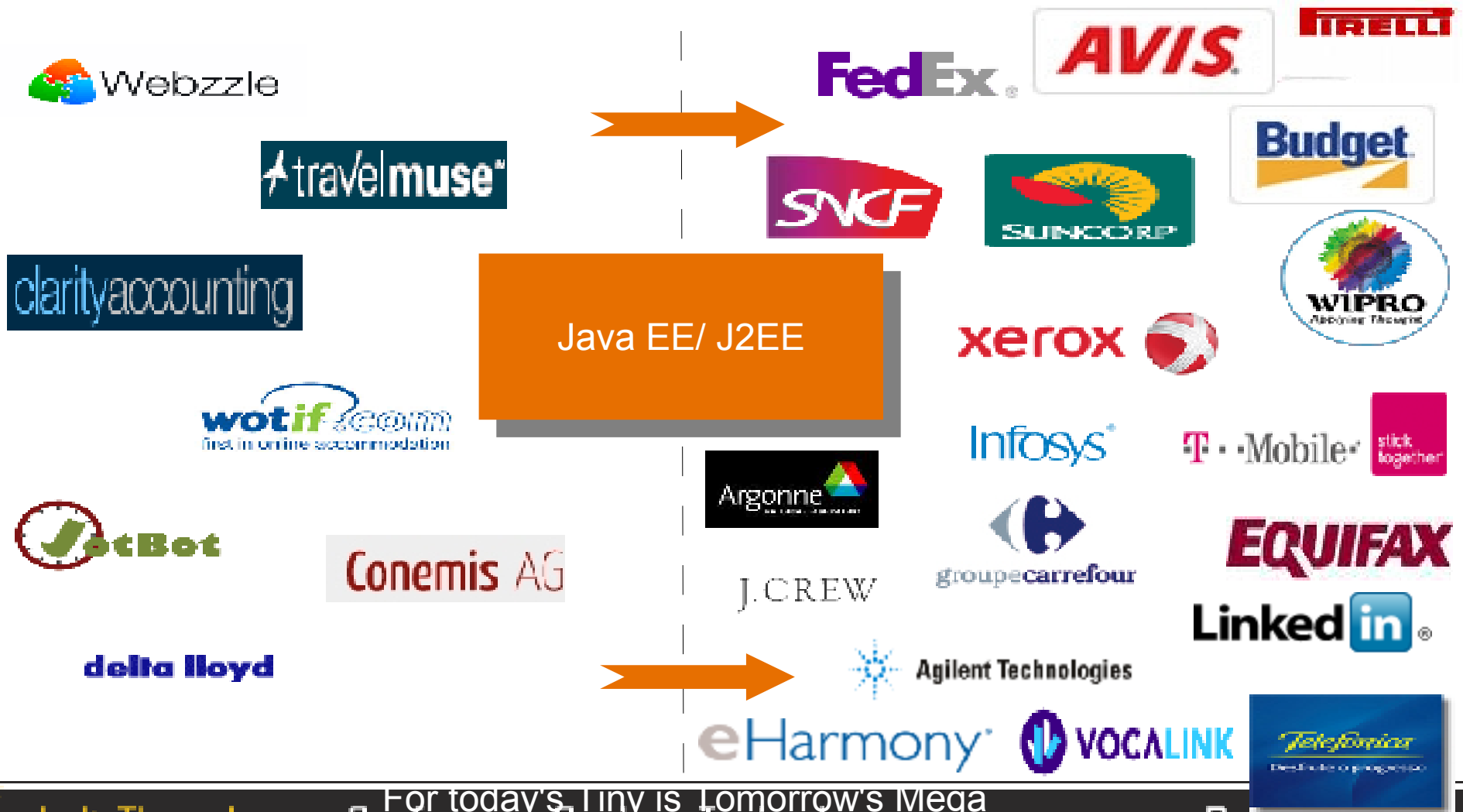
Over 18M
Downloads
since FY'06



Y/Y Download Growth



Java EE: Scales as you scale



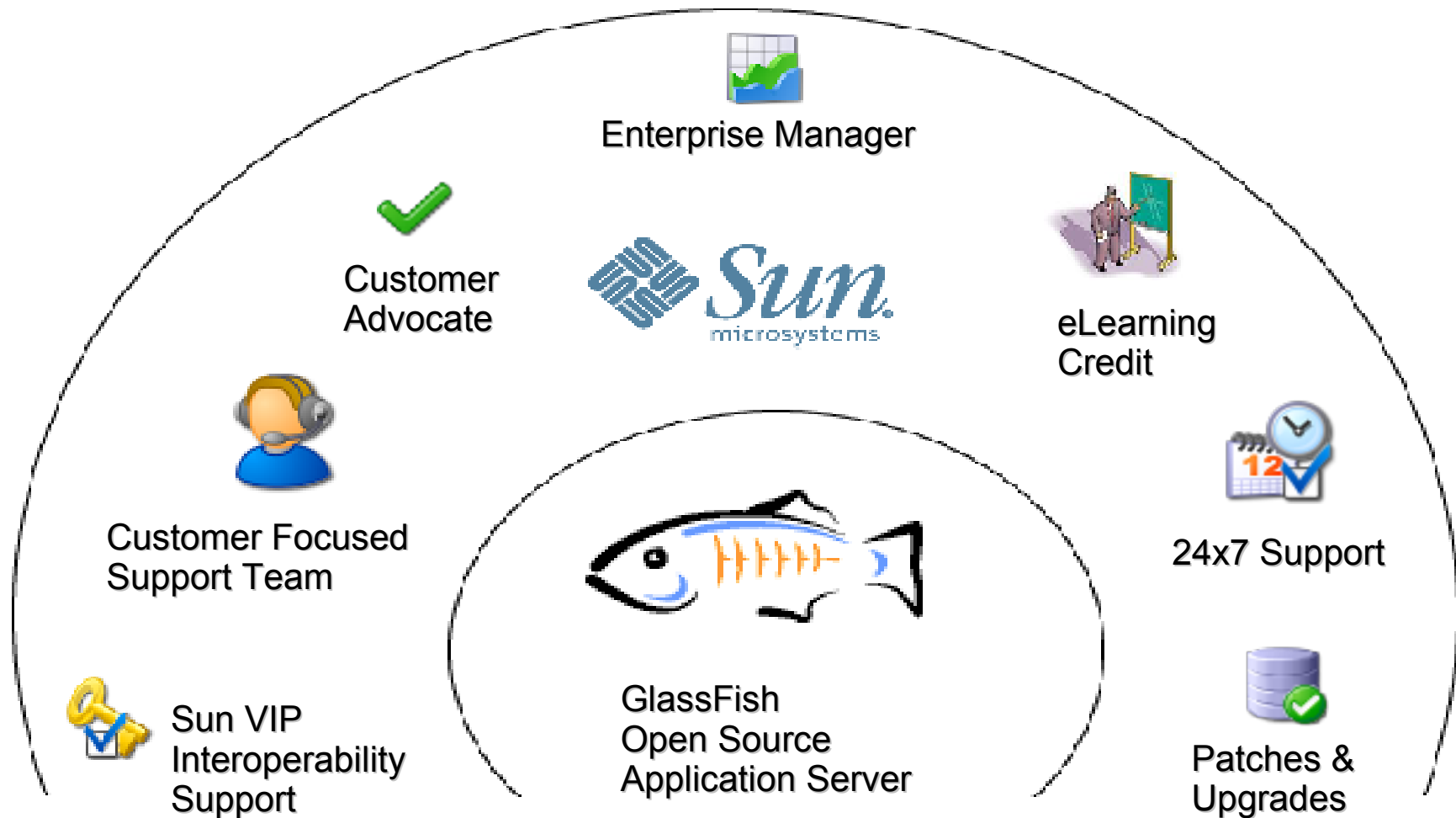
What is GlassFish ?



- A community
 - Users, Partners, Testers, Developers, ...
 - Started in 2005 on java.net
- Application Server
 - Enterprise Quality and Open Source (CDDL & GPL v2)
 - Java EE Reference Implementation
 - Full Commercial Support from Sun



Sun GlassFish Enterprise Server



GlassFish v3

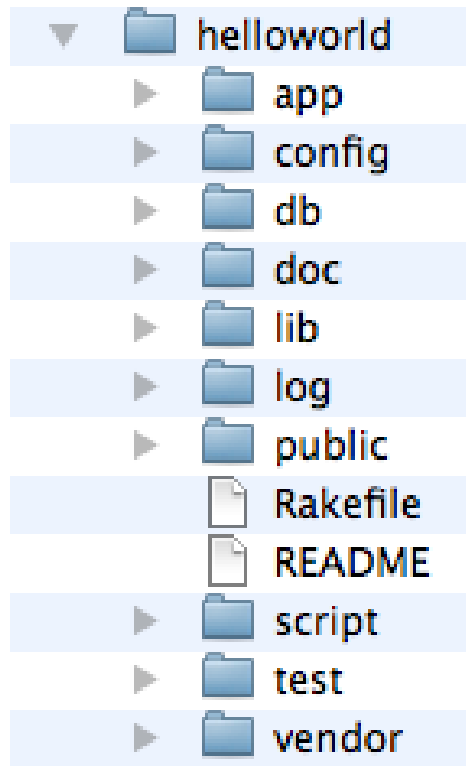
- Modular:
 - Maven 2 – Build & Module description
 - Felix – OSGi runtime (216 bundles)
 - Allow any type of Container to be plugged
 - Start Container and Services on demand
- Embeddable: runs in-VM
- Extensible
 - Rails, Grails, Django, ...

Dynamic Languages & Frameworks



<http://glassfish-scripting.dev.java.net>

Rails Deployment Choices



Credits: <http://birdwatchersdigest.com>

Demo

NetBeans / Eclipse & Java EE 6

http://blogs.sun.com/arungupta/entry/screencast_27_simple_web_application

http://blogs.sun.com/arungupta/entry/screencast_28_simple_web_application

http://blogs.sun.com/arungupta/entry/screencast_26_develop_run_debug/

http://blogs.sun.com/arungupta/entry/totd_93_getting_started_with/

http://blogs.sun.com/arungupta/entry/totd_94_a_simple_java

http://blogs.sun.com/arungupta/entry/totd_95_ejb_3_1

http://blogs.sun.com/arungupta/entry/totd_102_java_ee_6

http://blogs.sun.com/arungupta/entry/totd_99_creating_a_java

<http://blog.arungupta.me/2008/11/screencast-28-simple-web-application-using-eclipse-and-glassfish-v3-prelude/>



Embeddable GlassFish

```
public void testServlet() throws Exception {  
    int port = 9999;  
    GlassFish glassfish = newGlassFish(port);  
    URL url = new URL("http://localhost:" + port + "/" + NAME +  
"/SimpleServlet");  
    BufferedReader br = new BufferedReader(  
        new InputStreamReader(  
            url.openConnection().getInputStream()));  
    assertEquals("Wow, I'm embedded!", br.readLine());  
    glassfish.stop();  
}
```

Embeddable GlassFish

```
private GlassFish newGlassFish(int port) throws Exception {  
    GlassFish glassfish = new GlassFish(port);  
    ScatteredWar war = new ScatteredWar(NAME,  
        new File("src/main/resources"),  
        new File("src/main/resources/WEB-INF/web.xml"),  
        Collections.singleton(new  
File("target/classes").toURI().toURL()));  
    glassfish.deploy(war);  
    System.out.println("Ready ...");  
    return glassfish;  
}
```

CLI-based Administration

- “asadmin” CLI utility
- Administrative commands can be added with each container :

```
@Service(name="myCommand")
public class ChangeRandomCtr implements AdminCommand {
    @Param
    String s1;
    @Param
    String s2;
    ... }
```

–Available as :

```
asadmin myCommand -s1 foo -s2 bar
```

GlassFish™ v3 Administration Console

Tree

Common Tasks

- Registration
- GlassFish News
- Enterprise Server
- Applications
- Lifecycle Modules
- Resources
 - JDBC
 - Connectors
 - Resource Adapter Configs
 - JMS Resources
 - JavaMail Sessions
 - JNDI
- Configuration
 - JVM Settings
 - Logger Settings
 - Web Container
 - EJB Container
 - Ruby Container
 - Java Message Service
 - Security

Registration

GlassFish News

Subscriptions

Deployment

List Deployed Applications

Deploy an Application

Quick Start Guide

Administration Guide

Developer's Guide

Application Deployment Guide

Update Center

Installed Components

Available Updates

Available Add-Ons

Other Tasks

Create New JDBC Connection Pool



Run GlassFish Faster

Download the white paper and learn how to optimize GlassFish performance in a production environment. > [Get It Now](#)

- [Get production support.](#)
- [Stay current with GlassFish news at The Aquarium](#) or [learn about GlassFish deployments.](#)
- [Get community support, Developer Expert Assistance](#) or [Expert-to-engineer web training for Java EE 5.](#)
- [Join Project GlassFish](#)
- [Learn about upcoming Java EE 6 Release in this webinar.](#)

Demo

GlassFish v3 Administration

Light Weight & On-demand Monitoring

- Event-driven light-weight and non-intrusive monitoring
- Modules provide domain specific probes (monitoring events)
 - EJB, Web, Connector, JPA, Jersey, Orb, Ruby
- End-to-end monitoring on Solaris using DTrace
- 3rd party scripting clients
 - JavaScript to begin with

Demo

GlassFish v3 Monitoring

http://blogs.sun.com/arungupta/entry/totd_104_glassfish_v3_monitoring



REST Interface

- REST interface to management and monitoring data
 - Configuration data, Commands invocation (start/stop instance, deploy, undeploy, ...), CRUD resources (JMS, JDBC, ...)
 - localhost:4848/management/domain
 - localhost:4848/monitoring/domain
- GET, POST, DELETE methods
- XML, JSON, HTML reps

Demo

GlassFish v3 REST Interface

http://blogs.sun.com/arungupta/entry/totd_113_javafx_front_end

http://blogs.sun.com/arungupta/entry/totd_116_glassfish_v3_administration



GlassFish ISV Partners



References

- glassfish.org
- sun.com/glassfish
- blogs.sun.com/theaquarium
- twitter.com/glassfish
- glassfish@sun.com



Java EE 6 & GlassFish v3

Paving the path for future

11, 12 Dec. 2009

Pune, India

IndicThreads.com Conference On Java Technology

Arun Gupta (blogs.sun.com/arungupta, @arungupta)

Sun Microsystems, Inc.