



11:05-12:00 – GF v2 Overview - Karen & Eduardo

12:00-1 – Rich & Jonathan

**1-1:30 – Lunch Break**

1:30-2:25 - GF in Real Life - Peerflix, Wotif, Harvard U

1:30-2:25 - jMaki, Scripting - (web 2.0 track)

2:35-3:30 – GF Partners - Interface21, Caucho, Terracotta, Genuitec

4:00-4:55 – *on your own...*

5:05-6:00 – Gfv3 and OpenDS/OpenID

6:00-7:30 – Reception



## Demo Stations

- OpenDS/SSO/ID
- jMaki/Phobos/GFv3
- OpenESB
- OpenPortal
- GFv2 Admin/Monitoring
- GFv2 Clustering

## Partner Stations

- Interface21 (Spring)
- Genuitec (myEclipse)
- RedHat (Jboss/JSF)
- Jvantage
- Webtide (Jetty)
- Caucho (Quercus/PHP)



# Welcome to GlassFish Day and Overview of GlassFish v2

**Karen Tegan Padir**  
**Eduardo Pelegri-Llopart,**  
**Sun Microsystems, Inc**





# Agenda

- Welcome and Introductions
- JavaEE 5
- GlassFish Community
- GlassFish v2
- Previews of Next Sessions
- The Rich and Jonathan Show...



# **Karen Tegan Padir**

## **VP, Java Enterprise Software**

### **Sun Microsystems, Inc.**



# Eduardo Pelegri-Llopart Distinguished Engineer

## Sun Microsystems, Inc.



# Eduardo Pelegri-Llopart Distinguished Engineer

## Sun Microsystems, Inc.





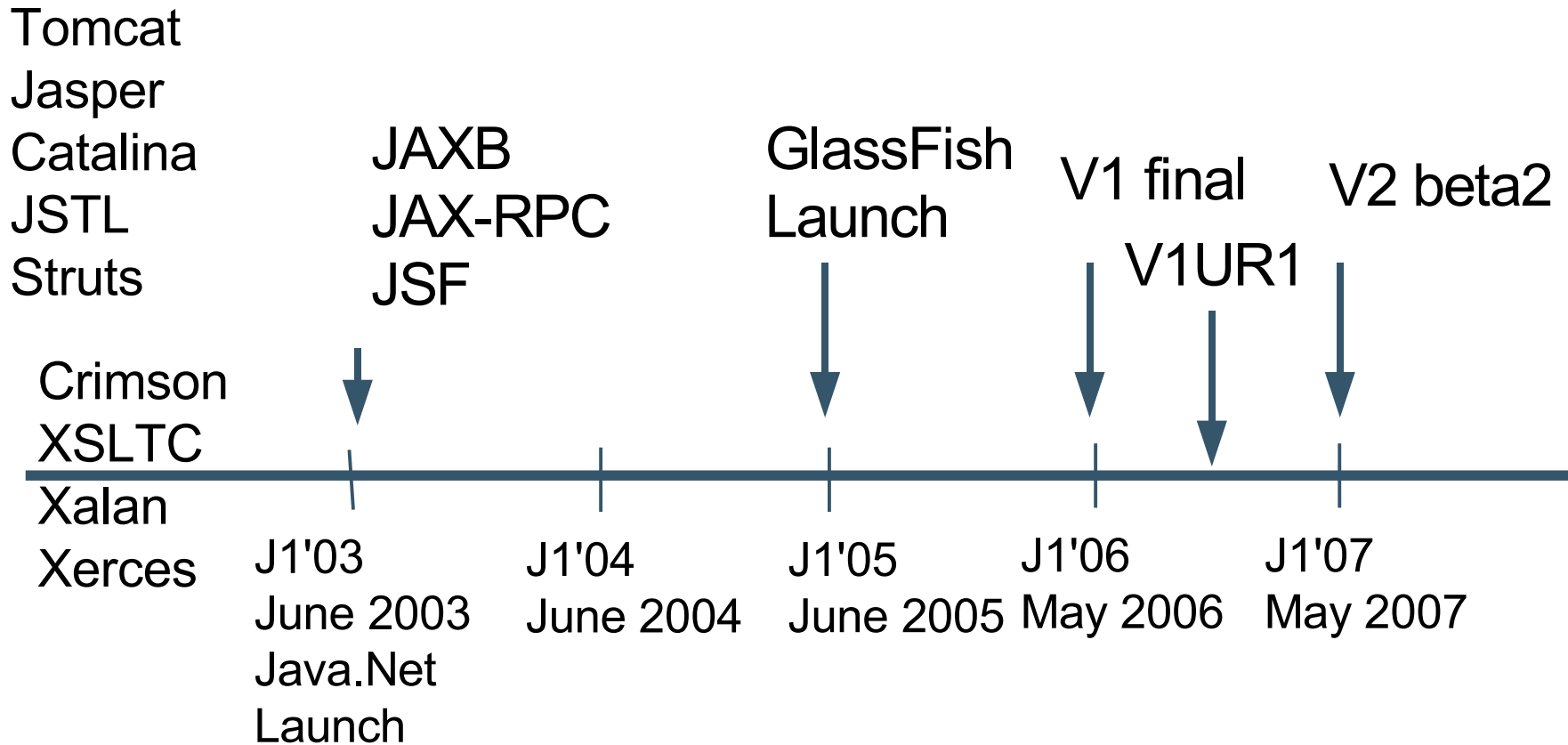
## For More Information

- <http://blogs.sun.com/theaquarium>
- <http://blogs.sun.com/stories>
- <http://glassfish.java.net>





# Timeline of Project GlassFish





# What Is Project GlassFish?

- Java EE 5 Reference Implementation
  - Included in Java EE 5 SDK
- Enterprise Quality
  - SJS AS 9.0 & 9.1 PE / SE
- Open Source
  - OSI license - CDDL (& GPLv2)
- Community at Java.Net
  - Sources, bug DBs, discussions, Roadmaps, Docs
- Expanding to Additional Projects
  - jMaki, Phobos, BlogApps, JAX-RS, Slynkr...



# ***A Taste of Java EE 5***



## J2EE 1.4

- J2EE is enormously powerful
  - The industry standard for robust enterprise apps
- But that power sometimes gets in the way
  - Too difficult to get started
  - Even simple apps need boring boilerplate
- Can we keep the power...  
but make typical development tasks simpler?
- **YES...** and that is the focus of Java EE 5!



# Java EE 5.0 = (J2EE 1.4).next

- Java EE 5 Theme: Ease of Development
- POJO-based programming
  - More freedom, fewer requirements
- Extensive use of annotations
  - Reduced need for deployment descriptors
  - Annotations are *the law*
- Resource Injection
- New APIs and frameworks



# Specification Changes in Java EE 5

- JAX-WS 2.0 & JSR 181
- Java Persistence
- EJB 3.0
- JAXB 2.0
- JavaServer Faces 1.2 – new to Platform
- JSP 2.1 – Unification w/ JSF 1.2
- StAX – Pull Parser – new to Platform

# Using JAX-RPC 1.1

```
package endpoint;
import java.rmi.*;
```

```
public class HelloServiceImpl
    implements HelloServiceSEI {

    public String sayHello(String param)
        throws java.rmi.RemoteException {
        return "Hello " + param;
    }
}
```

```
package endpoint;
import java.rmi.*;
```

```
public interface HelloServiceSEI
    extends java.rmi.Remote {
    public String sayHello(String param)
        throws java.rmi.RemoteException;
}
```

```
<?xml version='1.0' encoding='UTF-8' ?>
<webservices xmlns='http://java.sun.com/xml/ns/j2ee' version='1.1'>
  <webservice-description>
    <webservice-description-name>
      HelloService</webservice-description-name>
    <wsdl-file>
      WEB-INF/wsdl/HelloService.wsdl</wsdl-file>
    <jaxrpc-mapping-file>
      WEB-INF/HelloService-mapping.xml
    </jaxrpc-mapping-file>
    <port-component xmlns:wsdl-port_ns='urn:HelloService/wsdl'>
      <port-component-name>HelloService</port-component-name>
      <wsdl-port>wsdl-port_ns:HelloServiceSEIPort</wsdl-port>
      <service-endpoint-interface>
        endpoint.HelloServiceSEI</service-endpoint-interface>
      <service-impl-bean>
        <servlet-link>WSServlet_HelloService</servlet-link>
      </service-impl-bean>
    </port-component>
  </webservice-description>
</webservices>
<?xml version='1.0' encoding='UTF-8' ?>
<configuration
  xmlns='http://java.sun.com/xml/ns/jax-rpc/ri/config'>
  <service name='HelloService'
    targetNamespace='urn:HelloService/wsdl'
    typeNamespace='urn:HelloService/types'
    packageName='endpoint'>
    <interface name='endpoint.HelloServiceSEI'
      servantName='endpoint.HelloServiceImpl'>
    </interface>
  </service>
</configuration>
```

**OLD! J2EE 1.4** 



# Using JAX-WS 2.0

```
import javax.jws.WebService;
```

```
@WebService
```

```
public class MySimpleWS {
```

```
    public String sayHello(String s) {
```

```
        return "Hello " + s;
```

```
    }
```

```
}
```





# Stateless Session Bean in Java EE 5.0

```
import javax.jws.WebService;  
import javax.ejb.Stateless;
```

```
@WebService
```

```
@Stateless
```

```
public class MySimpleWS {  
    public String sayHello(String s) {  
        return "Hello " + s;  
    }  
}
```



# Java EE 5.0 Summary

- Java EE 5 Theme: Ease of Development
- Big Specification Improvements
- *Not your father's J2EE* – Bill Shannon



# The GlassFish Community



# What Is Project GlassFish?

- Java EE 5 Reference Implementation
  - Included in Java EE 5 SDK
- Enterprise Quality
  - SJS AS 9.0 & 9.1 PE / SE
- Open Source
  - OSI license - CDDL (like OpenSolaris, NetBeans)
- Community at Java.Net
  - Sources, bug DBs, discussions at Java.Net
  - Roadmaps, Architecture Documents



# Principles

- Industry-Leading Technology (no secret sauce)
- Want Adoption of Code Base
- Community Centered. Encourages Participation
- Transparency (bugs, roadmap); Clear Governance
- Continuous Improvements
- Participation (contributors, partners)
- Integrated and Integrateable
- Open Standards - JCP et al.
- OSI Licenses. CDDL (with a few others)
- Clear Copyright (SCA)



## Open Source and Sun

- Why?
  - Ubiquity, Lower Barrier to Adoption
  - Better Products, Closer to Customer
  - More Know-How, Contributors, Components
  - Larger, Faster adoption of Java EE 5
- Revenue?
  - Training, Support, Consulting. Systems
- Sun gets Big Portion of Pie; Pie Gets Bigger
  - Sun's (and others!) final take home is *bigger*



# Community Contributors

- Code donations
  - Sun Microsystems: SJS AS 8.x+
  - Oracle: Top Link Essentials
- Specification leadership
  - SUN, Oracle, BEA
- Participation from corporations
  - TmaxSoft, BEA, JBoss, Jetty
- Individual contributors



# The GlassFish AppServer



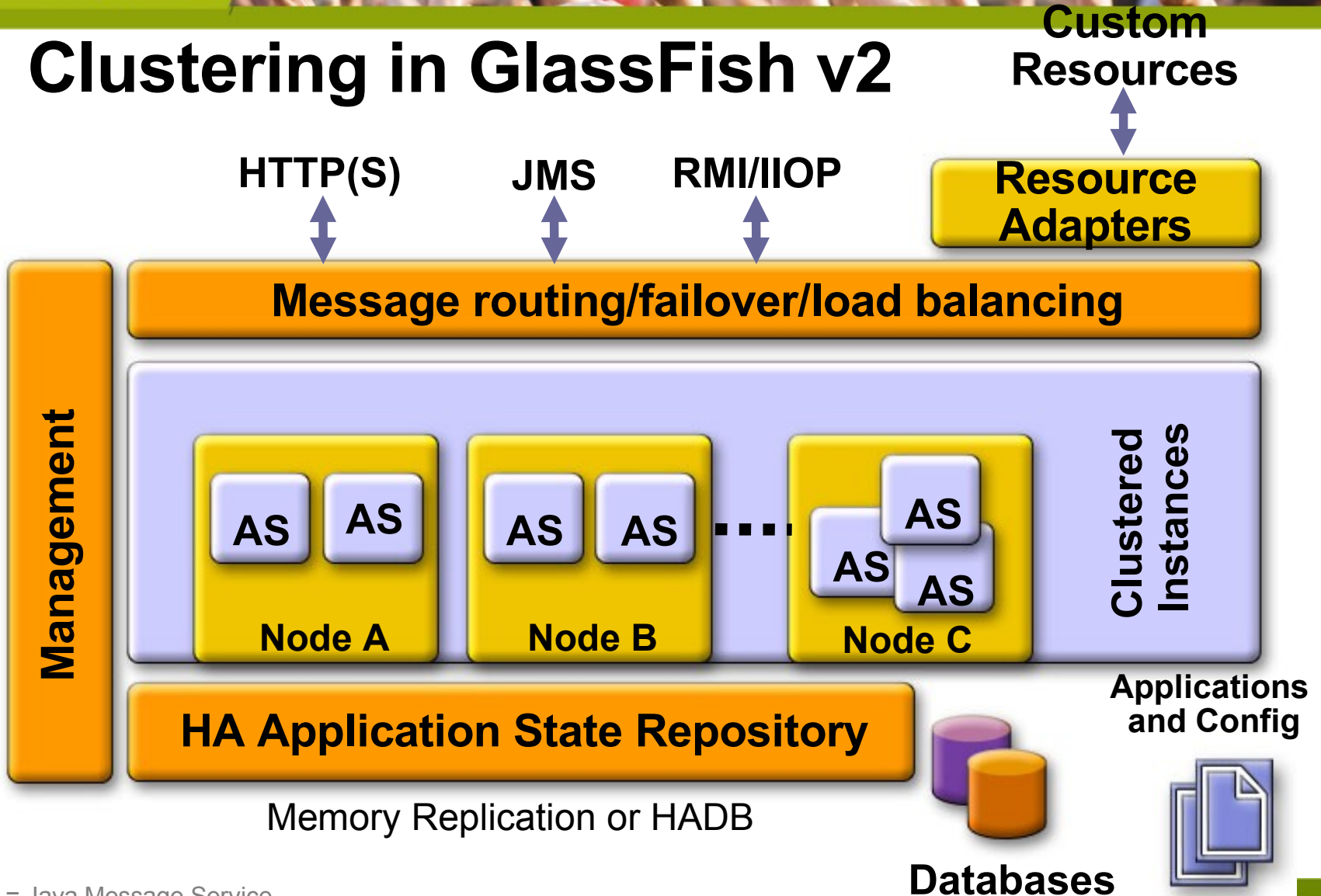


# Releases in Project GlassFish

- GlassFish v1
  - Released! – Victory! Java EE 5 Compliance!
  - UR1 - bug fixes
- GlassFish v2
  - New WS stack, performance, startup time
  - Load balancing, cluster management, failover
  - Some scripting support
  - Community, Transparency, Adoption
- GlassFish v3
  - Better modularization, better scripting support



# Clustering in GlassFish v2





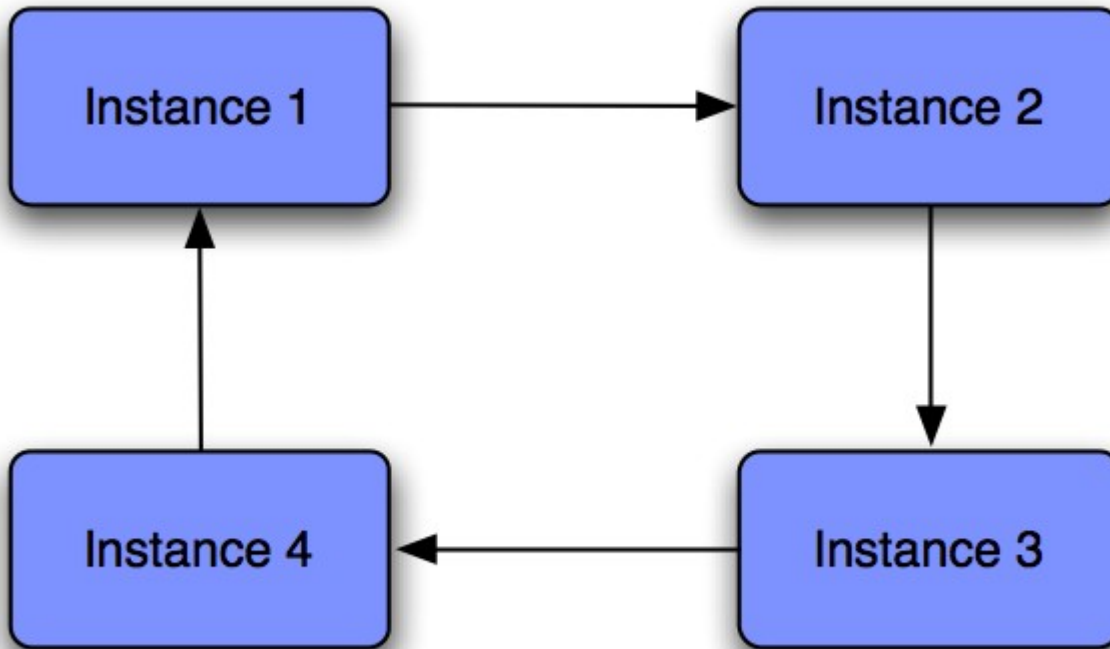
# Memory Replication

- HADB provides high availability for
  - Http session state
  - Stateful EJB session state
  - Single sign-on state
- HADB - still available; still right solution for some problems
  - enables proven “5 – nines” availability
  - more complex to administer
  - not (currently) open source
- Alternative: Memory Replication



# Memory Replication

## Typical Cluster Topology

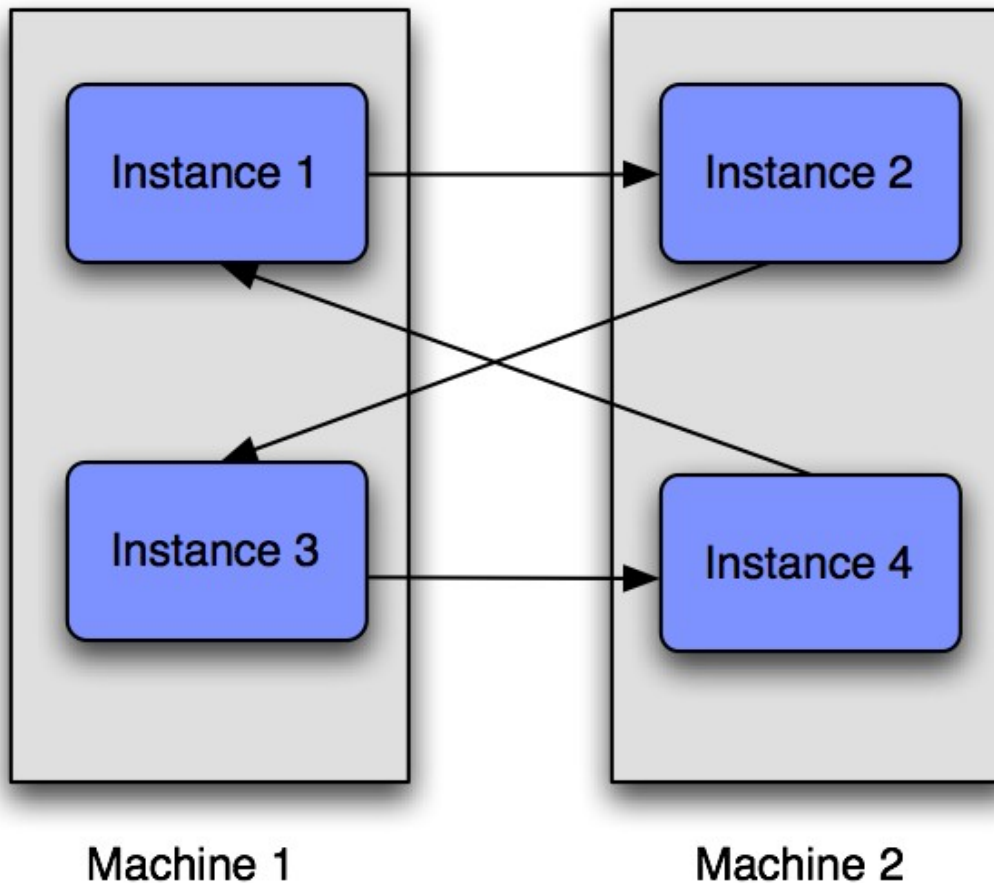




# Memory Replication

## Typical Cluster Topology

**Example: Maximize Availability on 4 node cluster on 2 machines**





# Memory Replication Configuration

## Out of the box ...

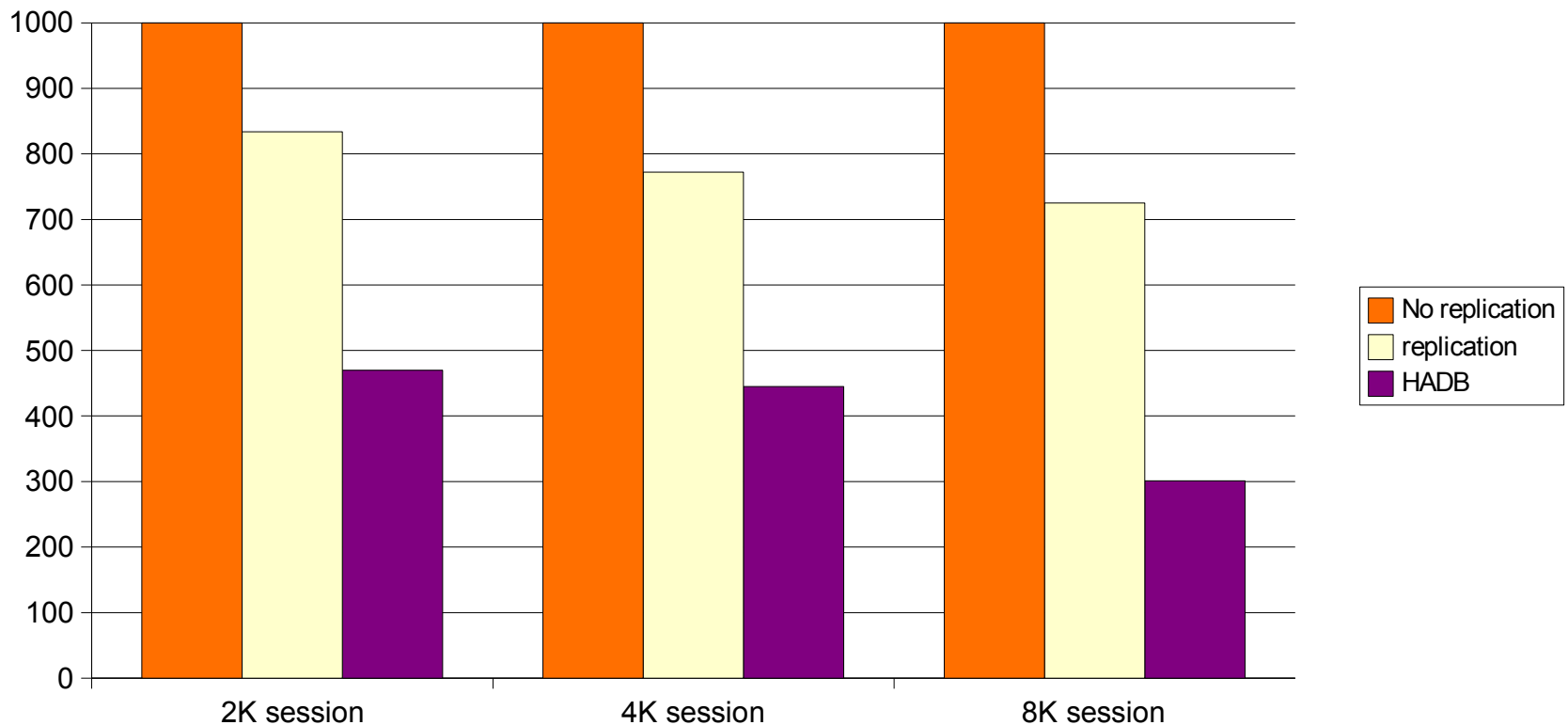
- Create a domain
  - Use the 'cluster' admin profile – defaults for replication are handled
    - Enables GMS – heartbeat enabled
    - persistence-type = “replicated”
- Create a cluster & instances
- Deploy your application with availability-enabled=true
- That's it



# Memory Replication Performance

## Cost of replication

Hits/Sec ~17-25% throughput degradation





# Memory Replication Implementation

Based on JXTA technology and Project Shoal/GMS

- JXTA technology
  - Not just for peer-to-peer!
  - Streamlined for high volume and throughput
  - Benefited from collaboration with Grizzly Project
- JXTA works well for us
  - Simplified api's made near zero-config story possible





# JAX-WS and JAXB Implementations

- Dynamic Runtimes
- Industry-Leading Implementations
  - JAXB RI is de-facto *Industry Standard*
  - JAX-WS is *much faster* than Axis
- WS Separation of Transport and Encoding
  - HTTP, JMS, SMTP, TCP/IP, Fast Infoset, SOAP, XML.
- *Project Tango/WSIT* – MS Interop.
- Growing # of extensions: JSON, Spring support
- **Great** Performance so far... & getting better



# Web Services Interoperability Technologies (WSIT)

- Enables Interop between Java and .NET 3.0
- Plugins to JAX-WS RI to provide:
  - Bootstrapping, optimizing communication, reliable messaging, atomic transactions, security and trust
  - WS-MEX, WS-Transfer, WS-Policy, WS-SecureConversation, WS-ReliableMessaging, WS-Coordination, WS-AtomicTransactions, WS-Security, WS-Trust
- Check TS-4865—“Takes Two to Tango...”



# Java Business Integration (JBI)

- Built in support for JBI (JSR -208)
- JBI runtime (OpenESB) and system components are bundled and integrated in GlassFish v2
  - <https://open-esb.dev.java.net>
- Administration GUI
- CLI enhanced for running JBI commands



# Usage Profiles

- Out of the box support for usage profiles
- Developer/administrator chooses profile at the time of domain creation
- Improve user experience based on the profile chosen
- GlassFish v2 will support
  - Developer profile
  - Cluster profile
  - Enterprise profile



# Security Enhancements

- JSR-196 (Java Authentication SPI for Containers)
- ECC (Elliptic Curve Cryptography) support
- Support for JKS (Java Key Store) format
- Support for “assign-groups” in Security Realm
- Support for JDBCRealm



# Web Container - JSP, Servlet, Grizzly

- Asynchronous Request Processing (ARP) and Comet
- Non-blocking SSL
- Apache AJP protocol
- In-memory (JSR 199) style JSP compilations
- All webcontainer aspects dynamically reconfigurable
- Alternate docroots for virtual servers and web apps
- Startup and request processing optimizations
- Nice adoption for Grizzly
  - Generic NIO framework (Grizzly 1.5)



# Top Link Essentials / JPA

- Oracle Contribution
- Very Active Community
  - Oracle, Sun, TmaxSoft, independents
  - Mail: [persistence@glassfish.dev.java.net](mailto:persistence@glassfish.dev.java.net)
- Pluggable
  - In GlassFish, JEUS, Jonas, Tomcat, Geronimo, Jboss, Oracle
- Converse is true also
  - Hibernate & Kodo / OpenJPA run on GlassFish



# Management Features

- Centralized, secure, remote access
  - Accessible as GUI, CLI, IDEs, Java-based programs
- Off-line configuration
- Industry-standard infrastructure
  - Java Management Extensions, JMX™ API
- Custom ANT tasks
- Management and Monitoring API
  - Application Server Management eXtensions, AMX
- Can be monitored through *jConsole* and others





# New Admin Console

- Admin console re-implemented using JSF Templating, JavaServer™ Faces technology for creating Pages and Components
- jMaki Charting
- Ajax (breadcrumbs, tree, restart)
- Support for Cluster Management
- One step Deployment with Server-side Browsing
- Integrated JBI Administration



# Specification Updates

- JSR 196 (Authentication SPI)
- JSR 208 (JBI)
- JavaServer Faces 1.2 MR
- JavaServer Pages™ (JSP™ pages) 2.1 MR
- JSP Standard Tag Library (JSTL) 1.2 MR
- Servlet 2.5 MR
- JAX-WS MR
- Java Architecture for XML Binding (JAXB) MR



# *Previews of* **Other Sessions Today**



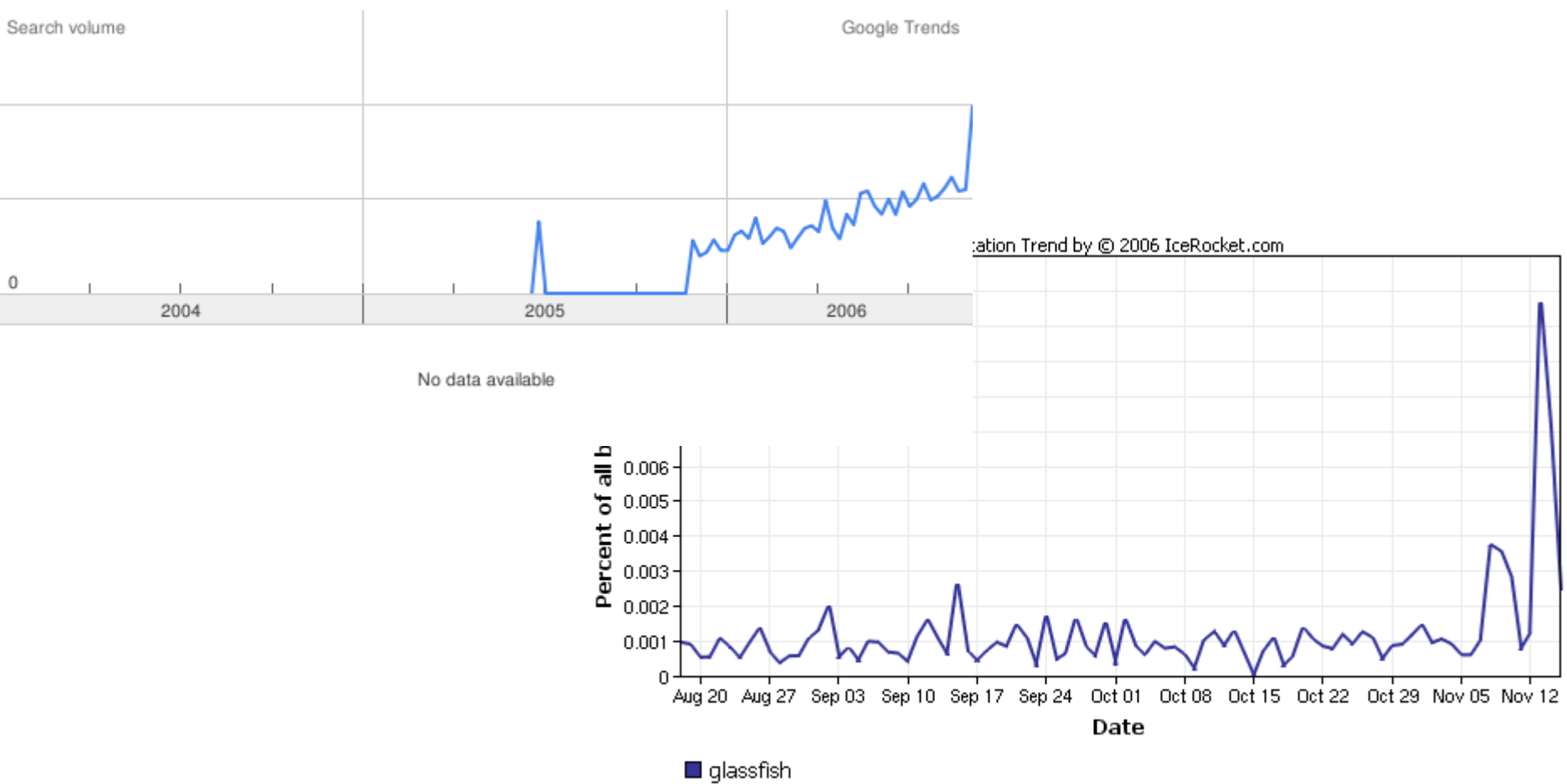
# AJAX and Scripting Activities

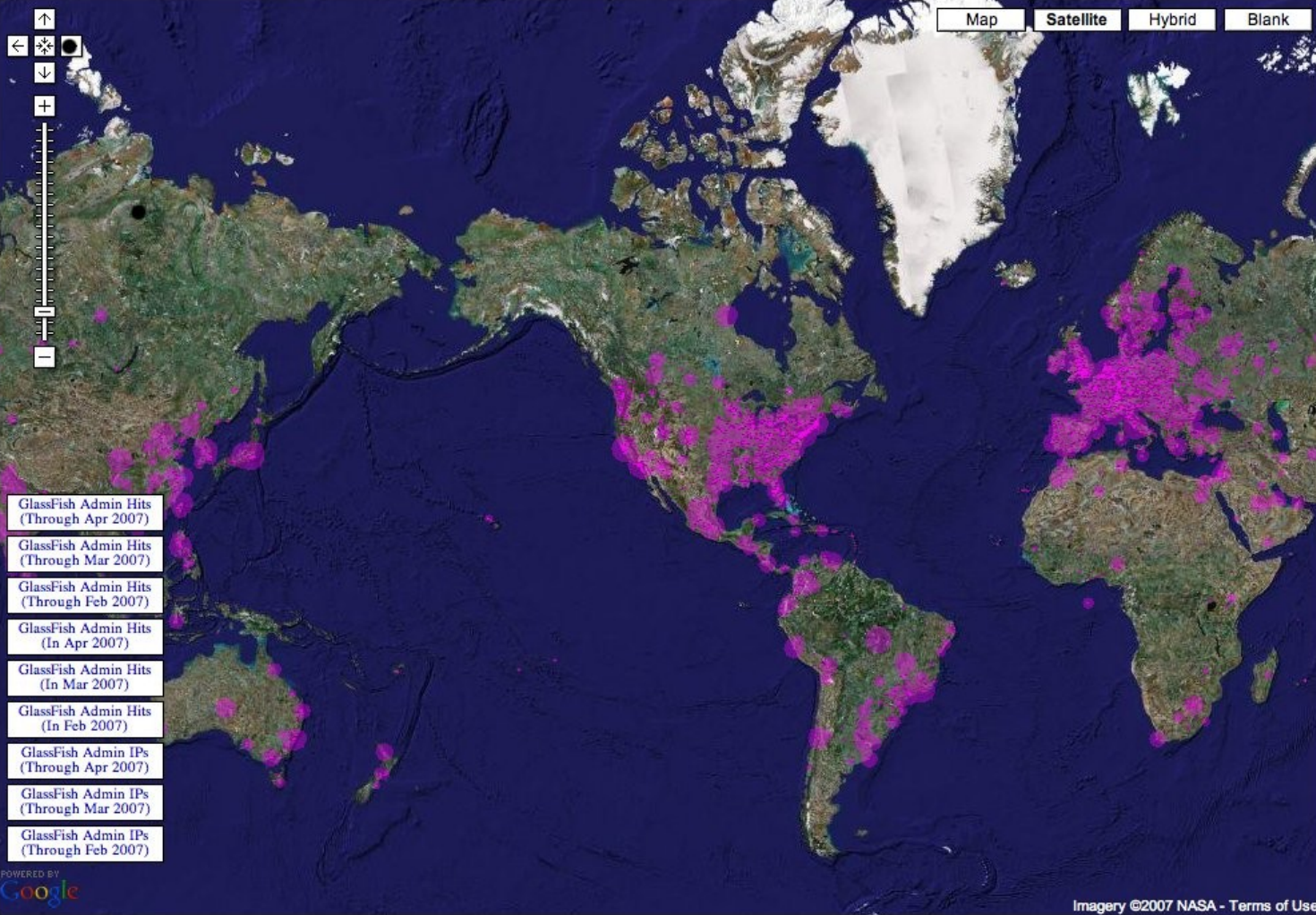
- jMaki - <http://ajax.dev.java.net>
  - Encapsulation of AJAX widgets
- Phobos - <http://phobos.dev.java.net>
  - Scripting on the client
- See Web 2.0 track session
- 
- JRuby – Ruby on a JVM
- Quercus – PHP on a JVM
  - See Caucho and v3 demo station



# Adoption Stories

- PeerFlix
- Wotif
- Harvard's Metaverse
  
- Check this afternoon's session
- <http://blogs.sun.com/Stories>





[Link To: Default View](#) | [This View](#)

Current Zoom: 2 (details at 14+)

[What is This Map?](#)

**Stats For Visible Area**

**GlassFish: 305260**





# GlassFish Partnership Program

- Interface21 (Spring)
- Genuitec (myEclipse)
- Webtide (Jetty)
- Caucho (Quercus)
- Noemax (.NET WS client)
- Terracotta (Clustering JVM)
  
- Talk with us... Just starting
- Check this afternoon's session





# Why GlassFish V3?

- Modular : you only load what you use
- Open : support all types of managed applications.
- Starts in 0.5s on the best PC, 0.7 on my laptop.
- Can run in a phone !
  - And will scale up to the enterprise!
- **GlassFish V3 is the meta-container for Web 2.0**
- See Last Session of this Track



# Q & A





11:05-12:00 – GF v2 Overview - Karen & Eduardo

12:00-1 – Rich & Jonathan

**1-1:30 – Lunch Break**

1:30-2:25 - GF in Real Life - Peerflix, Wotif, Harvard U

1:30-2:25 - jMaki, Scripting - (web 2.0 track)

2:35-3:30 – GF Partners - Interface21, Caucho, Terracotta, Genuitec

4:00-4:55 – *on your own...*

5:05-6:00 – Gfv3 and OpenDS/OpenID

6:00-7:30 – Reception



## Demo Stations

- OpenDS/SSO/ID
- jMaki/Phobos/GFv3
- OpenESB
- OpenPortal
- GFv2 Admin/Monitoring
- GFv2 Clustering

## Partner Stations

- Interface21 (Spring)
- Genuitec (myEclipse)
- RedHat (Jboss/JSF)
- Jvantage
- Webtide (Jetty)
- Caucho (Quercus/PHP)



# And now for something completely Different...



**Jonathan Schwartz**  
**CEO and President**

**Rich Green**  
**Executive VP, Software**

**Sun Microsystems, Inc.**