



GlassFish Partner Presentations

Patrick Dyson
Director of Engineering
Sun Microsystems



GlassFish Partner Presentations

Meet the Partners and their technologies

Learn about the GlassFish Partner Program and hear selected partners describe their technologies and integrations



Agenda

GlassFish Partner Program Overview

Genuitec – MyEclipse 5.5 and GlassFish Integration

Caucho – Quercus, PHP on Java

Interface21 – Spring!

TerraCotta – Clustering with DSO

Q&A for all at the end



GlassFish Partner Program

Building the business community

- Promote, distribute, build adoption of YOUR GlassFish related business
 - GlassFish Partner web page
 - Exposure and comarketing (like this)
 - Distribution through the Update Center
 - Channel for support
- OEM, Integrator, Technology, and Support
- Commitment to validate latest version
- Certification program in the works



For More Information

- GlassFish: <https://glassfish.dev.java.net>
- Partner: glassfish-partnerships@sun.com
- SPA: <http://sun.com/partners>



Agenda

GlassFish Partner Program Overview

Genuitec – MyEclipse 5.5 and GlassFish V2

Todd Williams VP-Technology, Genuitec LLC

Caucho – Quercus, PHP on Java

Interface21 – Spring!

TerraCotta – Clustering with DSO



MyEclipse 5.5: Introduction to GlassFish 2.0 and Java EE 5 Support

Todd Williams

VP-Technology, Genuitec LLC

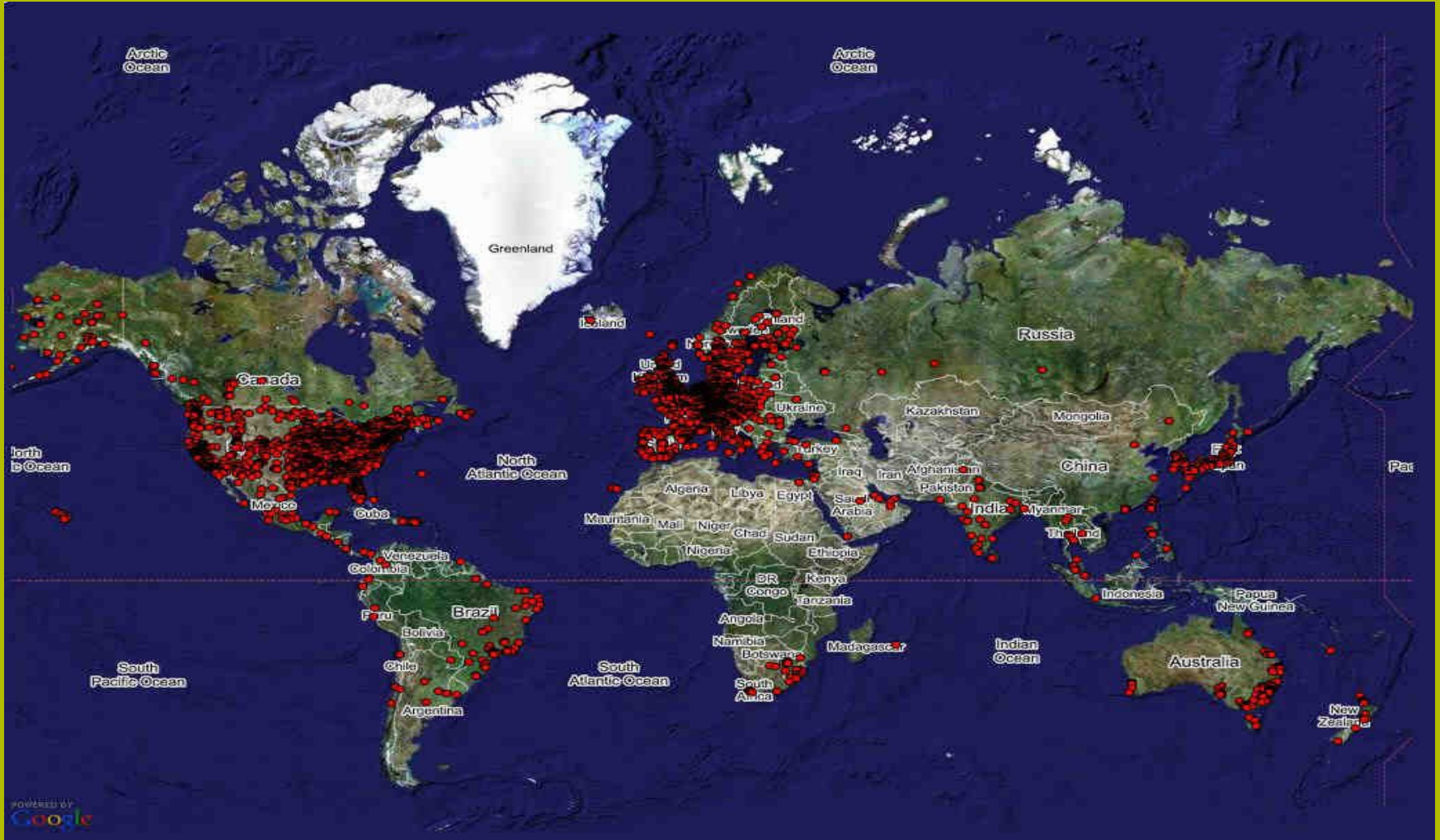


Introduction to Genuitec

- Founded in 1997
- Focus: Enhancing Developer Productivity
 - Products, Training, Consulting
- MyEclipse Enterprise Workbench
 - Java EE / J2EE / AJAX / XML / DB / UML
 - Most widely used enterprise IDE based on Eclipse
 - 500K users in 150 countries in 12K companies
 - First released in 2003 with 6-9 releases yearly
 - Subscription pricing: \$30-\$50 / year / developer



Where MyEclipse Users Live: Top 5000 Cities





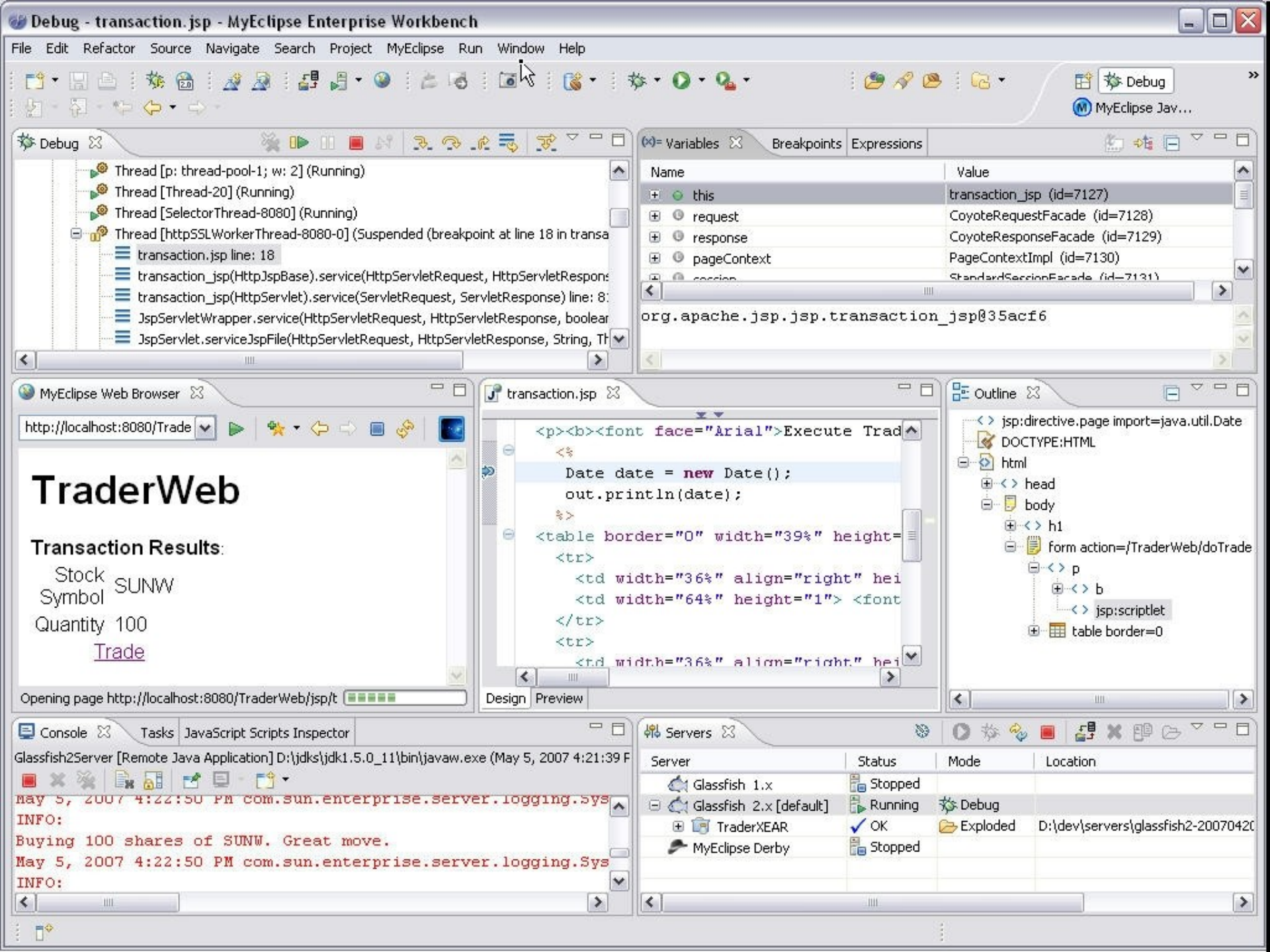
New in MyEclipse 5.5

- A ton of features I'm not going to talk about
- Java EE 5 Spec-compliant project creation
- EJB 3
 - Stateless, Stateful, Entity, Message-driven
- JPA
 - Smart editing and annotations support
 - Validation
 - Reverse-engineer entities and DAO's from a database
 - TopLink, OpenJPA, Hibernate persistence providers



New in MyEclipse 5.5 (cont)

- GlassFish V2
 - Control the server within the IDE
 - Start / Stop / Deploy / Undeploy
 - Minimal configuration, just point and debug
 - Spec-compliant archive deployments
 - EAR, WAR, EJB-jar
 - RAD exploded deployments
 - Hot-Sync debugging
 - Modifications are injected immediately upon save
 - Application keeps running – no restarts needed
- Availability: 5/14/2007-ish



TraderWeb

Transaction Results:

Stock Symbol SUNW
Quantity 100

[Trade](#)

```
<p><b><font face="Arial">Execute Trad  
<*>  
Date date = new Date();  
out.println(date);  
<*>  
<table border="0" width="39%" height=  
<tr>  
<td width="36%" align="right" hei  
<td width="64%" height="1"> <font  
</tr>  
<tr>  
<td width="36%" align="right." hei
```

Outline

- jsp:directive.page import=java.util.Date
- DOCTYPE:HTML
- html
 - head
 - body
 - h1
 - form action=/TraderWeb/doTrade
 - p
 - b
 - jsp:scriptlet
 - table border=0

Console

Glassfish25Server [Remote Java Application] D:\jdk1.5.0_11\bin\javaw.exe (May 5, 2007 4:21:39 F

May 5, 2007 4:22:50 PM com.sun.enterprise.server.logging.Sys

INFO:

Buying 100 shares of SUNW. Great move.

May 5, 2007 4:22:50 PM com.sun.enterprise.server.logging.Sys

INFO:

Servers

Server	Status	Mode	Location
Glassfish 1.x	Stopped		
Glassfish 2.x [default]	Running	Debug	
TraderXEAR	OK	Exploded	D:\dev\servers\glassfish2-20070420
MyEclipse Derby	Stopped		



Summary

- MyEclipse 5.5 – Build anything on any tier
 - Fantastic support for GlassFish V2 and Java EE 5
- GlassFish V2
 - Reference standard, fast development, great team
- Fusion Technology
 - Matisse4MyEclipse
 - NetBeans Use-anywhere connector for SNAPs
 - Visual HTML/JSP Designer, Database Explorer
 - Image Editor, Screen Capture, XML Designer
- Demos at the Genuitec pod



Q&A

- ...at the end...



Agenda

GlassFish Partner Program Overview

Genuitec – MyEclipse 5.5 and GlassFish V2

Caucho – Quercus, PHP on Java

Sam Heisz

Interface21 – Spring!

TerraCotta – Clustering with DSO



Caucho Technology, Inc.

www.caucho.com

Quercus PHP



Company Profile

- Founded in 1998
- Privately held
- Based in La Jolla, California
- Engineering focus
- Over 100 global resellers
- Java EE Licensee



About Quercus

- 100% Java implementation of the PHP language
- Servlet and javax.scripting, works with any application server (GlassFish)
- GNU Public license
- First public release in December, 2006, first release independent of Resin in December, 2006



Motivation

- PHP is a proven effective language for creating websites
- 28,475,056 domains using PHP (April 2006, source Netcraft)
- Web innovation in PHP (Ajax, Web 2.0)
- Availability of PHP developers
- Strong demand



Replacement for C implementation

- Thread safe
- Database connection pooling
- Extensible
- No pointer and buffer overruns
- Complete solution with no external dependencies
- Comprehensive testing of Quercus code



Web front end for Java

- Use with existing Java infrastructure (Java EE, Spring)
- Use for new development, leverage the strengths of PHP (website development), and Java (Java EE, Spring, javax.persistence, unit testing)
- Use and extend popular PHP packages (MediaWiki, Drupal, WordPress, etc)
- javax.script support (JSR 223)



Target Java solutions to PHP developers

- A new market for Java
- Enables Java tool vendors to target their Java products to the PHP market
- Enables open and closed source development of extensions to the PHP language



Extending PHP

- Quercus enables the use of Java objects and methods
- PHP programmers use PHP semantics
- Allows targeted implementation or reimplementations of functionality
- Objects can exist beyond the lifetime of a request
- Objects can be shared amongst requests



Response

- Very high level of interest
- Scalability and reliability
- Take advantage of existing expertise, bring together PHP and Java groups within the organization
- PHP enables rapid response to markets and business opportunities



Community

- Website quercus.caucho.com
- Point releases, snapshots, Subversion repository
- Quercus forum at forum.caucho.com
- Quercus bugs at bugs.caucho.com
- Patches applied and integrated into Caucho QA process
- Module uploads



Summary

- 100% implementation of the PHP library with thread safety
- A replacement for the C implementation of PHP
- A new market for Java
- GPL license
- www.caucho.com
- Pavilion booth #1332



Agenda

GlassFish Partner Program Overview

Interface21 – Spring!

Genuitec – MyEclipse 5.5 and GlassFish V2

Caucho – Quercus, PHP on Java

Interface21 – Spring!

Rod Johnson

TerraCotta – Clustering with DSO



Interface21 – *Spring from the Source*

- The company behind the Spring Framework
- Founded by core Spring team
- Sustains and drives Spring forward
- Interface21 services
 - Production and Development Support for Spring
 - Training
 - Thousands of developers trained in public and private trainings
 - Mentoring and High Level Consulting



***Much* more than an IoC container...**

Core component model

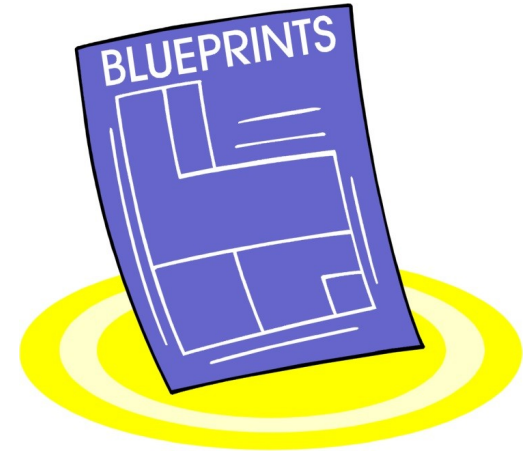
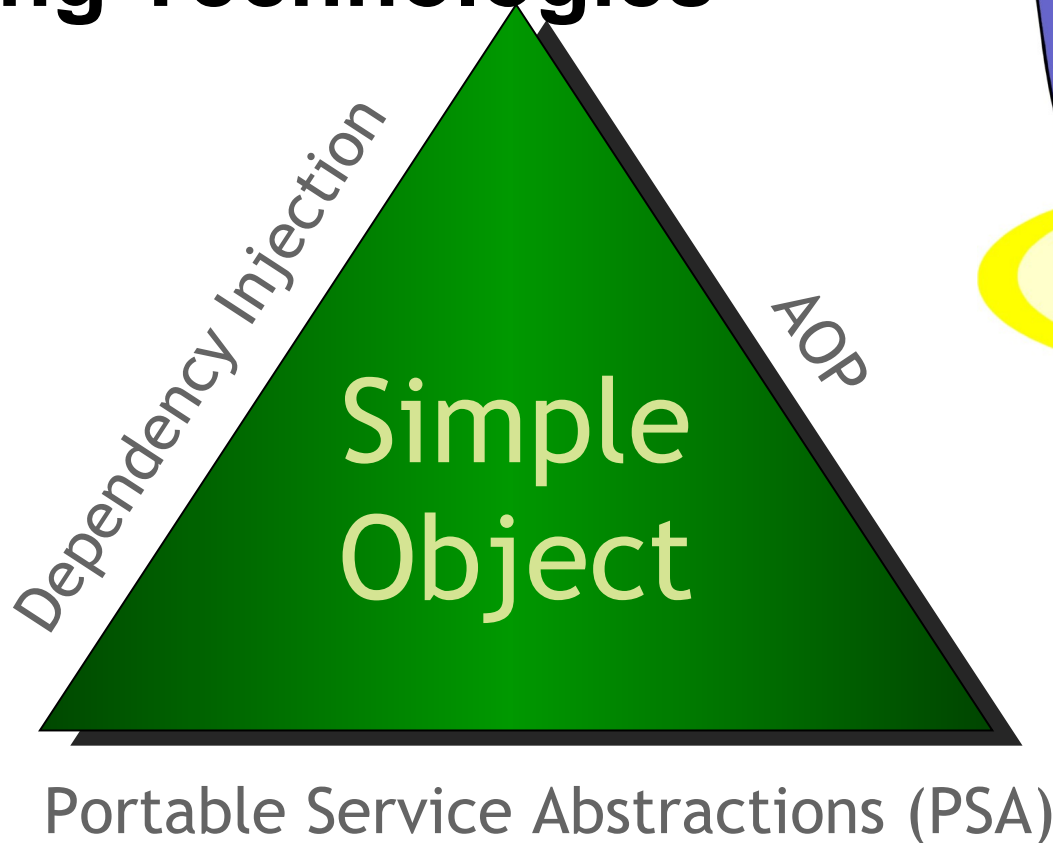
- + Services
- + Patterns (Recipes)
- + Integration (Ecosystem)
- + Portability (Runs everywhere)

= Universal POJO programming model

- Embraced by more and more enterprise vendors



Enabling Technologies





Not just about the Spring Framework: The Spring Portfolio

- Spring Framework
 - Spring WebFlow
 - Spring Web Services
 - Spring Security
 - Spring Rich Client
 - Spring LDAP
 - Spring IDE
 - Spring OSGi
 - Spring Modules
 - Spring.NET
 - Pitchfork
- AspectJ





What's New for Java One 2007?

- Spring 2.1M1
 - Allows use of annotations for configuration, as well as XML
 - Can mix and match annotations and XML
 - JCA 1.5 support
 - Further enhancements in JPA support
- Aims: Make Spring still easier to use



Java code: Annotations autoscanned

@Component

```
public class FooServiceImpl
    implements FooService {
    @Autowired
    private FooDao fooDao;

    public String foo(int id) {
        return fooDao.findFoo(id);
    }
}
```

-
-
-

@Aspect

```
public class ServiceInvocationCounter {
    private int useCount;

    @Pointcut("execution(* demo.FooService+.*(..))")
    public void serviceExecution() {}

    @Before("serviceExecution()")
    public void countUse() {
        this.useCount++;
    }

    public int getCount() {
        return this.useCount;
    }
}
```



Bootstrap configuration

- ```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:context="http://www.springframework.org/schema/context"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-2.0.xsd
http://www.springframework.org/schema/context
http://www.springframework.org/schema/context/spring-context-2.1.xsd">

<context:annotation-scan base-package="demo"/>

<bean id="fooDaoImpl" class="demo.FooDaoImpl">
<property name="dataSource" ref="dataSource"/>
</bean>

<bean id="dataSource" class="org.springframework.jdbc.datasource.DriverManagerDataSource">
<property name="driverClassName" value="org.hsqldb.jdbcDriver"/>
<property name="url" value="jdbc:hsqldb:mem:hsqldb:foo"/>
<property name="username" value="sa"/>
</bean>

</bean>
```



# What's New for Java One 2007?

- Spring Web Flow 1.1M1
  - **Significantly enhanced integration with JSF**
  - **Conversation-scoped persistence contexts**
- On the 1.1 final roadmap (August)
  - Acegi Security integration
  - Flow reuse: inheritance and composition
  - Java based flows



# Spring and GlassFish

- JAX-WS RI uses Spring as an extension point
  - Classes exposed as endpoints can be configured using Spring
  - Handlers and other artifacts can be configured using Spring
  - Brings Spring IoC, AOP and out of the box services to any remote endpoint
    - Includes Spring 2.0 custom scopes and @AspectJ support
- Other integrations likely in future



# JAX-WS RI Spring Support

- BookService will need to access other objects
  - Data access objects, audit logger, etc
- How does BookService and MyHandler talk to each other?

```
@WebService
class BookService {
 List<Book> getRecommendedBooks () {
 ...
 }
}
class MyHandler implements Handler<Message> {
 int affiliateId;
 ...
}
```



# JAX-WS RI Spring Support

## Sample configuration

```
<beans xmlns=...>
 <bean id="myHandler" class="foo.MyHandler" />
```

```
MyHandler {
 ...
}
```



# JAX-WS RI Spring Support

## Sample configuration

```
<beans xmlns=...>
 <bean id="myHandler" class="foo.MyHandler" />

 <bean id="myService" class="foo.BookService">
```

```
MyHandler {
```

```
 ...
```

```
}
```

```
BookService {
```

```
 ...
```

```
}
```



# JAX-WS RI Spring Support

## Sample configuration

```
<beans xmlns=...>
 <bean id="myHandler" class="foo.MyHandler" />

 <bean id="myService" class="foo.BookService">
 <property name="handler" ref="myHandler" />
 </bean>
</beans>
```

```
MyHandler {
 ...
}

BookService {
 MyHandler handler;
}
```





# JAX-WS RI Spring Support

## Sample configuration

```
<beans xmlns=...>
 <bean id="myHandler" class="foo.MyHandler" />

 <bean id="myService" class="foo.BookService">
 <property name="handler" ref="myHandler" />
 </bean>

 <wss:binding url="/stockQuote">
 <wss:service>
 <ws:service bean="myService">
 <ws:handlers>
 <ref bean="myHandler" />
 </ws:handlers>
 </ws:service>
 </wss:service>
 </wss:binding>
</beans>
```



# JAX-WS RI Spring Support

## Sample configuration

```
<beans xmlns=...>
 <bean id="myHandler" class="foo.MyHandler" />

 <bean id="myService" class="foo.BookService">
 <property name="handler" ref="myHandler" />

 <wss:binding url="/stockQuote">
 <wss:service>
 <ws:service bean="myService">
 <ws:handlers>
 <ref bean="myHandler" />
 </ws:handlers>
 </ws:service>
 </wss:service>
 </wss:binding>
</beans>
```



# JAX-WS RI Spring Support

## Sample configuration

```
<beans xmlns=...>
 <bean id="myHandler" class="foo.MyHandler" />

 <bean id="myService" class="foo.BookService">
 <property name="handler" ref="myHandler" />
 </bean>

 <wss:binding url="/stockQuote">
 <wss:service>
 <ws:service bean="myService">
 <ws:handlers>
 <ref bean="myHandler" />
 </ws:handlers>
 </ws:service>
 </wss:service>
 </wss:binding>
</beans>
```



# JAX-WS RI Spring Support

## Sample configuration

```
<beans xmlns=...>
 <bean id="myHandler" class="foo.MyHandler" />

 <bean id="myService" class="foo.BookService">
 <property name="handler" ref="myHandler" />

 <wss:binding url="/stockQuote">
 <wss:service>
 <ws:service bean="myService">
 <ws:handlers>
 <ref bean="myHandler" />
 </ws:handlers>
 </ws:service>
 </wss:service>
 </wss:binding>
</beans>
```



## To Learn More

- Spring Framework Web site
  - [www.springframework.org](http://www.springframework.org)
- Interface21 Web site
  - [www.interface21.com](http://www.interface21.com)
- Interface21 blogs
  - [blog.interface21.com](http://blog.interface21.com)



# Agenda

GlassFish Partner Program Overview

Interface21 – Spring!

Genuitec – MyEclipse 5.5 and GlassFish V2

Caucho – Quercus, PHP on Java

Interface21 – Spring!

**TerraCotta – Clustering with DSO**

Sreeni Iyer Senior Manager of Field Engineering



# *Terracotta – General Purpose Java Clustering*

**Sreeni Iyer,**  
**Senior Manager of Field Engineering,**  
**Terracotta @ <http://www.terracotta.org>**



# Terracotta DSO Technology

Learn how Terracotta DSO (Distributed Shared Objects) Technology allows for declarative clustering of JVM-state across a cluster of JVMs





# Agenda

Demo – JTable

High Level Architecture Overview



# DEMO

JTable Demo:

2 (n) plain Java Swing apps with “Model” clustered.

Enter a servlet deployed to a GlassFish Container.

State is shared amongst the Java apps and the servlet.



# Code Sample

```
package demo.jtable;
import javax.swing.JFrame;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.table.DefaultTableModel;

class TableDemo extends JFrame {

 private DefaultTableModel model; // Shared object
 private static Object[][] tableData = {
 { " 9:00", "", "", ""}, { "10:00", "", "", ""}, { "11:00", "", "", ""},
 { "12:00", "", "", ""}, { " 1:00", "", "", ""}, { " 2:00", "", "", ""},
 { " 3:00", "", "", ""}, { " 4:00", "", "", ""}, { " 5:00", "", "", ""}
 };

 TableDemo() {
 super("Table Demo");
 setSize(350, 220);
 setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 Object[] header = {"Time", "Room A", "Room B", "Room C"};
 model = new DefaultTableModel(tableData, header);
 JTable schedule = new JTable(model);
 getContentPane().add(new JScrollPane(schedule), java.awt.BorderLayout.CENTER);
 }

 public static void main(String[] args) {
 new TableDemo().setVisible(true);
 }
}
```

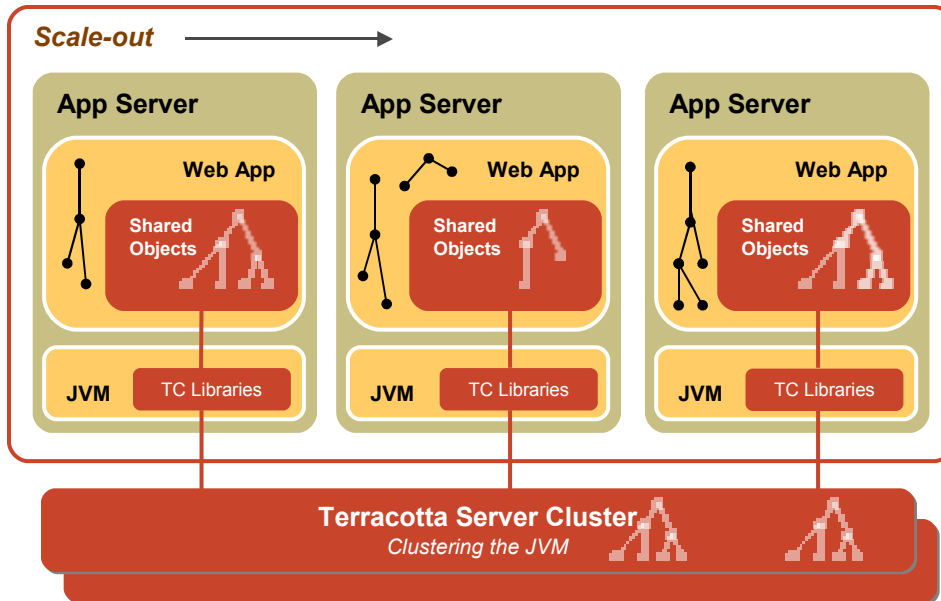


# Supporting Xml Config

```
<roots>
 <root>
 <field-name>demo.jtable.TableDemo.model</field-name>
 <root-name>ModelRoot</root-name>
 </root>

 <root>
 <field-name>demo.jtable.Roots.model</field-name>
 <root-name>ModelRoot</root-name>
 </root>
</roots>
```

# Terracotta Core Features



## Heap Level Replication

- Declarative, No Serialization
- Fine Grained / Field Level
  - GET\_FIELD( object, fieldname );
  - PUT\_FIELD( object, fieldname, value);
- Only Where Resident

## JVM Coordination

- Distributed Synchronized Block. Wait/Notify, Method Invocation.
- Fine Grained Locking
  - MONITOR\_ENTRY();
  - MONITOR\_EXIT();

- Large Virtual Heaps
- Management Console



# Summary

- DSO provides a declarative, very efficient mechanism to cluster state (any arbitrary set of POJOs - e.g. session, cache, POJOs, Spring Beans etc. ) across JVMs participating in a cluster.
- Developers gain simplicity (write POJOs) and faster time-to-market, since clustering gets delivered as a service of the run-time.
- Ops gains from improved application availability, “add-a-node” scale and improved cluster manageability.



## For More Information

Visit Terracotta's Booth at Java One.

Visit <http://www.terracotta.org> (Forums, Mailing Lists, Docs)

Visit <http://www.terracottatech.com> (Services, Training, Support)

Checkout Sessions on

Tuesday (TS-88040: Distributed Computing) by Ari/Jonas.

Wednesday (TS-6410: DWR) by Geert.

Friday (TS-1326: BCI) by Eugene/Tim.



# Q&A

## Caucho, Terracota, Genuitec, and Interface21