



Main sponsor



# GlassFish OSGi - from modular runtime to hybrid applications

Alexis Moussine-Pouchkine

Picasso

Do you really get  
Memory?

Jevgeni Kabanov

Matejko + Canaletto

Where is my table now?  
On adapting Java and  
SQL code to ever-changing  
database model.

Tomasz Skutnik

Malczewski + Chelmonski

Rich Grails UIs with  
Canoo RIA Suite

Dierk König

# Java Enterprise Edition (Java EE)

- Scalable and maintainable enterprise apps
- Container vs. Developer
  - Life-cycle, resource injection, security, ...
- Portable code
  - Portable skills
  - Vendor independence
- Industry standard
  - large developer community,
  - books, training, best practices, consultants, ...



# Java EE 6

- New features
  - Web profile, REST, managed beans, standard dependency injection, validation
- Enhanced APIs
  - EJB 3.1, JSF 2.0, JPA 2.0, Servlet 3.0
- More ease of use
  - Optional XML, simpler packaging, extensibility
- Usable as is
  - The end of the 100's MB .war

# GlassFish Community

- Trusted by developers
  - Over 10 million downloads annually
  - 50K active users in 30 days w/GlassFish Server 3
  - Active user forums
  - Sub-projects: Jersey, Metro, Grizzly, ...
- Application Server
  - Open Source and high-quality runtime
  - Java EE 5 / 6 Reference Implementation, early access to latest standards
  - Full Commercial Support from Oracle
  - Now with full cluster (GlassFish 3.1, Feb 2011)

# Enterprise Software

- Expectations
  - Deal with complexity and age
  - Ability to modify part of a (running) system
- Best Practices
  - Make your system modular
  - Define and document system interfaces

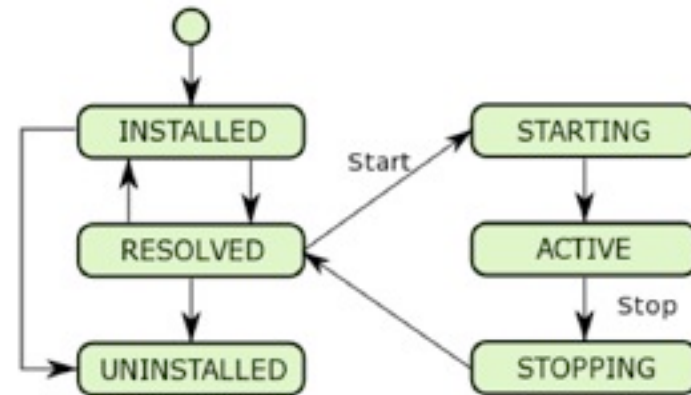
# Enterprise Software Reality

- Java offers a **classpath** as the only means of dealing with dependencies
- Java EE provides a limited set of modularity conventions
- Software versions are implicit and not enforced
- Java public really means “public”
- When best practices aren't enough...



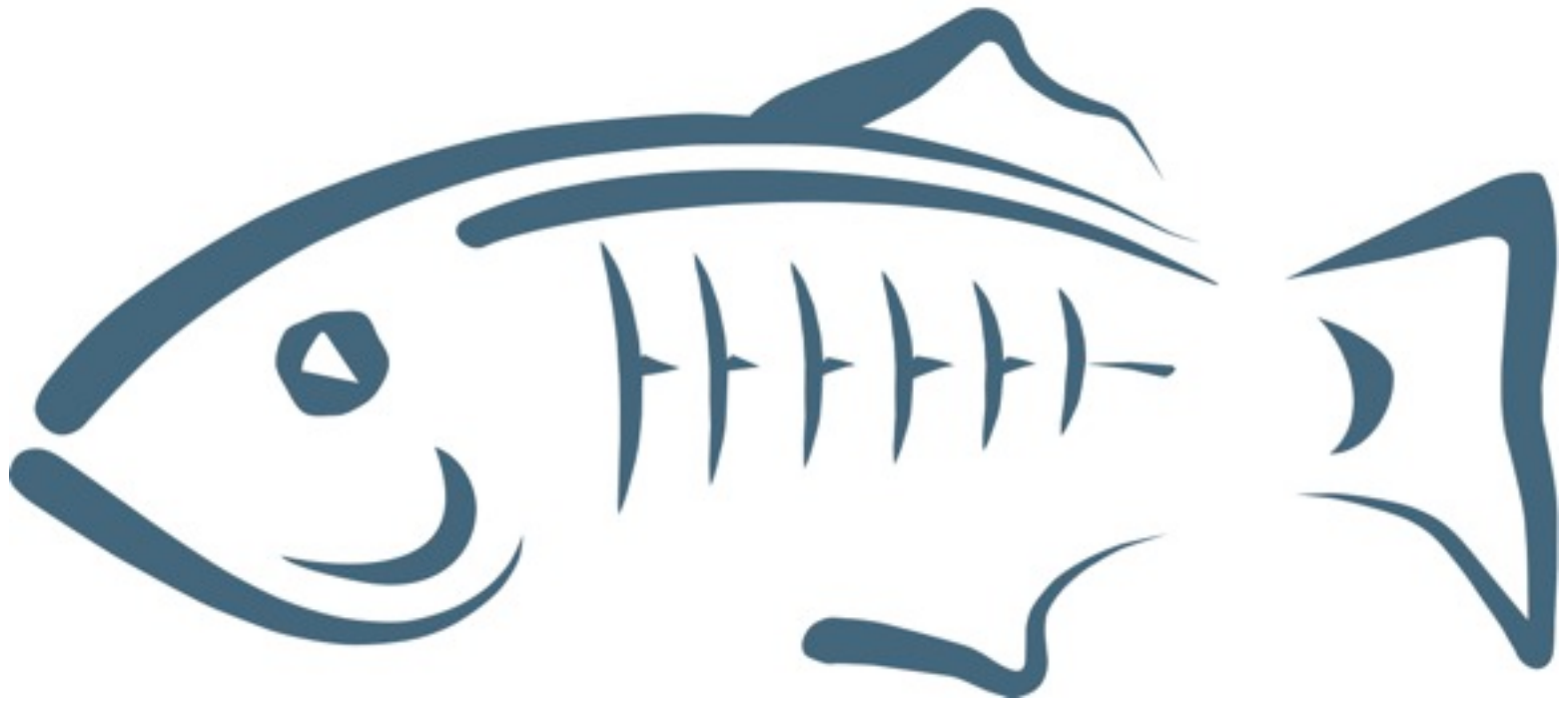
# OSGi 101

- Module = “Bundle” = JAR with additional metadata (import/export packages, ...)
- 5 states of an OSGi bundle
  - INSTALLED
  - RESOLVED
  - STARTING
  - ACTIVE
  - STOPPING



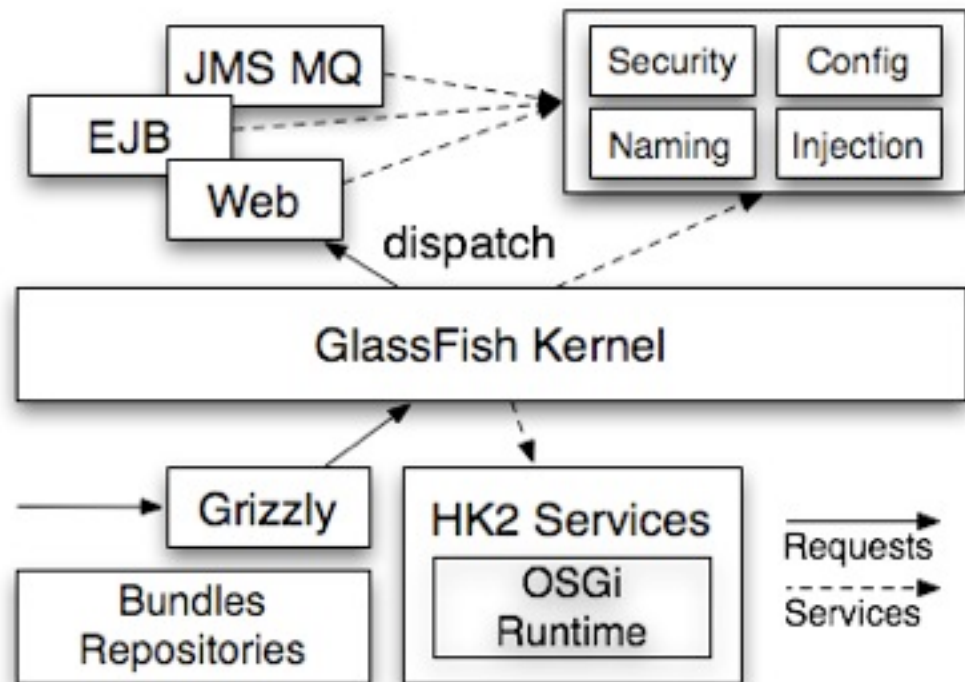


# Introducing GlassFish 3



# Modular and Dynamic

- Modular : Apache Felix (OSGi)
- Extensible : HK2
- Yet very Fast !



## GlassFish™ Server Open Source Edition

## Tree

## Common Tasks

- Registration
- GlassFish News
- Enterprise Server
- Applications
- Lifecycle Modules

## Resources

- JDBC
  - JDBC Resources
  - Connection Pools
    - \_\_TimerPool
    - DerbyPool
- Connectors
- Resource Adapter Configs
- JMS Resources
- JavaMail Sessions
- JNDI
- Configuration
  - JVM Settings
  - Logger Settings
  - Web Container
  - EJB Container
  - Ruby Container
- Java Message Service
- Security

## General

## Advanced

## Additional Properties

## Edit JDBC Connection Pool

Save

Modify an existing JDBC connection pool. A JDBC connection pool is a group of reusable connections for a particular d

Load Defaults

Flush

Ping

\* Indicates n

## General Settings

Pool Name: DerbyPool

Resource Type: javax.sql.DataSource

Must be specified if the datasource class implements more than 1 of the interface.

Datasource Classname: org.apache.derby.jdbc.ClientDataSource

Vendor-specific classname that implements the DataSource and/or XADataSource APIs

Driver Classname:

Vendor-specific classname that implements the java.sql.Driver interface.

Ping: ☐ Enabled

When enabled, the pool is pinged during creation or reconfiguration to identify and warn of a erroneous values for its attributes

Description:

## Pool Settings

Initial and Minimum Pool Size: 8 Connections

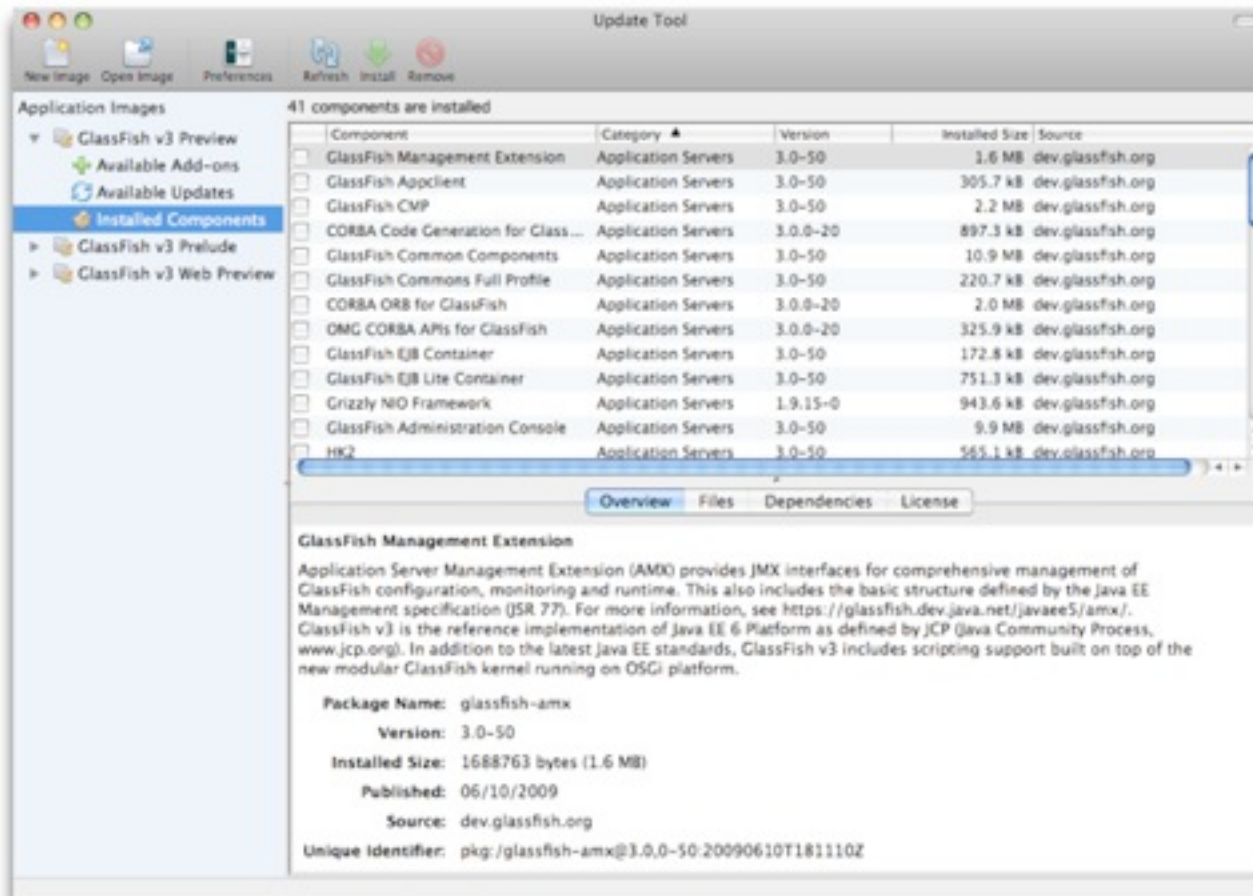
Minimum and initial number of connections maintained in the pool

Maximum Pool Size: 32 Connections

Maximum number of connections that can be created to satisfy client requests

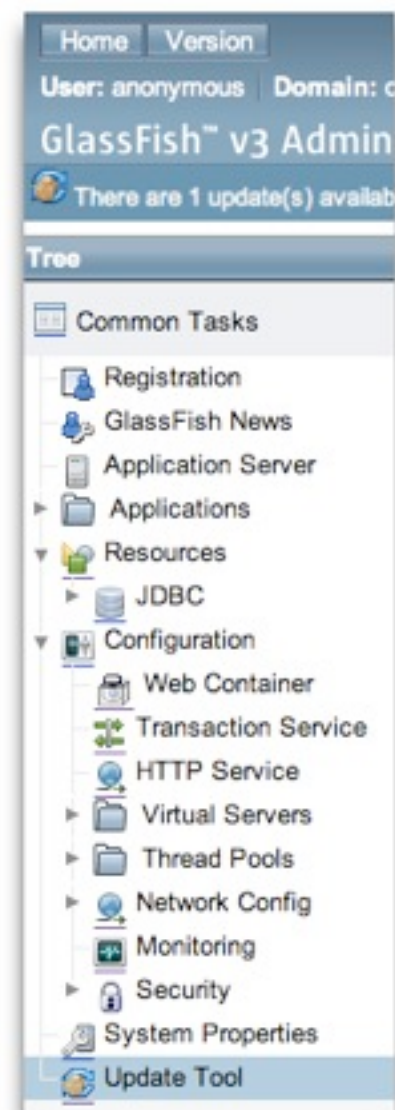
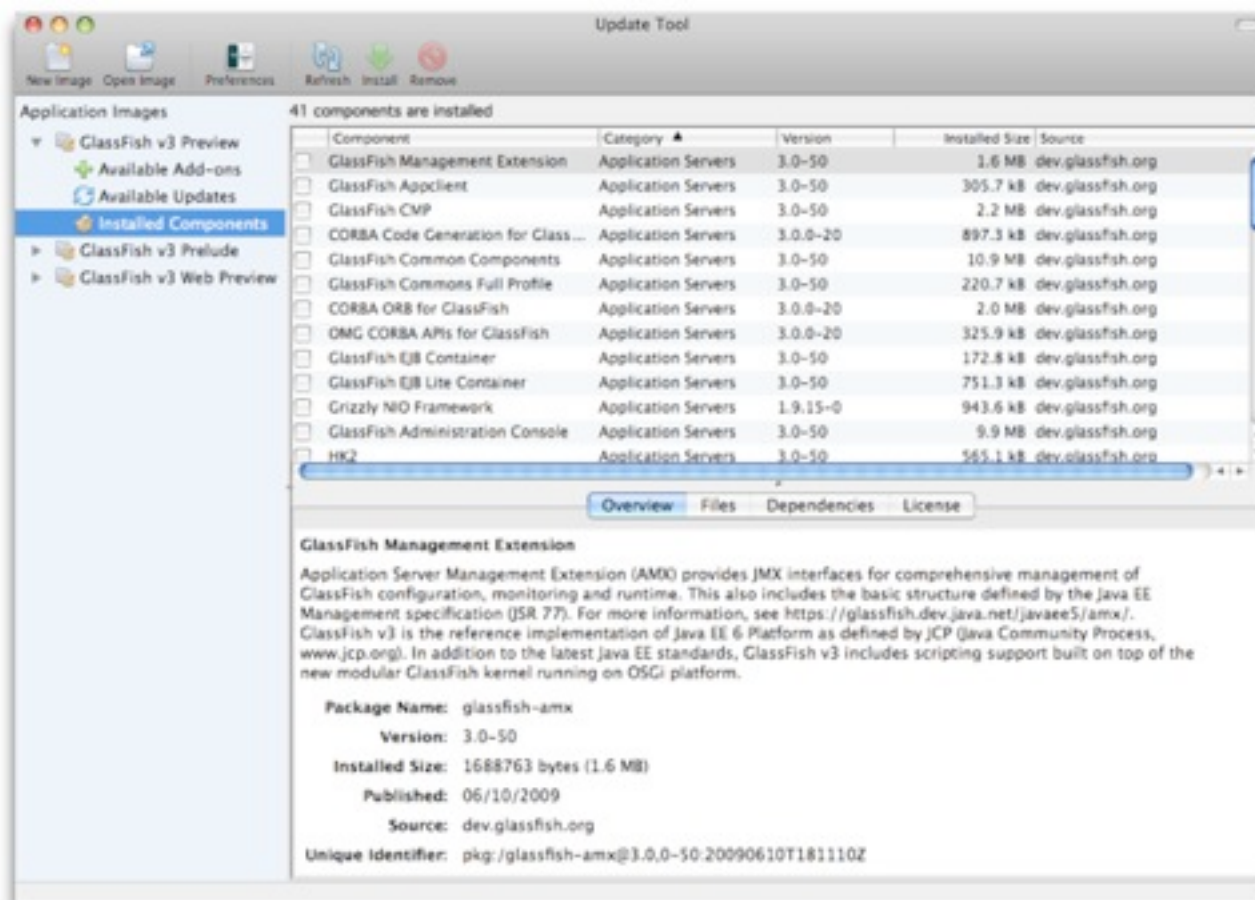
Pool Resize Quantity: 2 Connections

# Update Center





# Update Center



# Update Center

The screenshot shows the GlassFish v3 Update Tool interface. The top bar includes buttons for 'New Image', 'Open Image', 'Preferences', 'Refresh', 'Install', and 'Remove'. The left sidebar shows 'Application Images' with options like 'GlassFish v3 Preview', 'Available Add-ons', 'Available Updates', 'Installed Components' (selected), 'GlassFish v3 Prelude', and 'GlassFish v3 Web Preview'. The main area displays a table of 41 installed components.

Component	Category	Version	Installed Size	Source
GlassFish Management Extension	Application Servers	3.0-50	1.6 MB	dev.glassfish.org
GlassFish Applclient	Application Servers	3.0-50	305.7 kB	dev.glassfish.org
GlassFish CMP	Application Servers	3.0-50	2.2 MB	dev.glassfish.org
CORBA Code Generation for Class...	Application Servers	3.0.0-20	897.3 kB	dev.glassfish.org
GlassFish Common Components	Application Servers	3.0-50	10.9 MB	dev.glassfish.org
GlassFish Commons Full Profile	Application Servers	3.0-50	220.7 kB	dev.glassfish.org
CORBA ORB for GlassFish	Application Servers	3.0.0-20	2.0 MB	dev.glassfish.org
OMG CORBA APIs for GlassFish	Application Servers	3.0.0-20	325.9 kB	dev.glassfish.org
GlassFish EJB Container	Application Servers	3.0-50	172.8 kB	dev.glassfish.org
GlassFish EJB Lite Container	Application Servers	3.0-50	751.3 kB	dev.glassfish.org

Below the table, a terminal window shows the output of the command `% pkg list`:

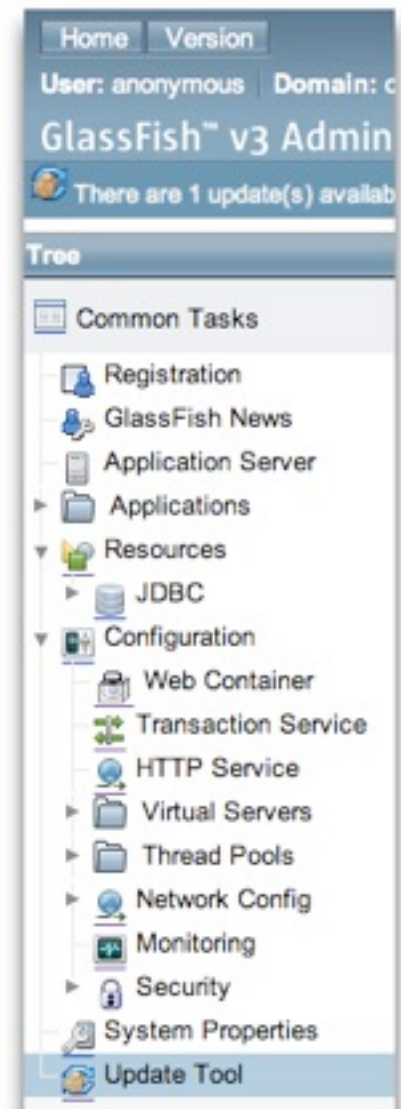
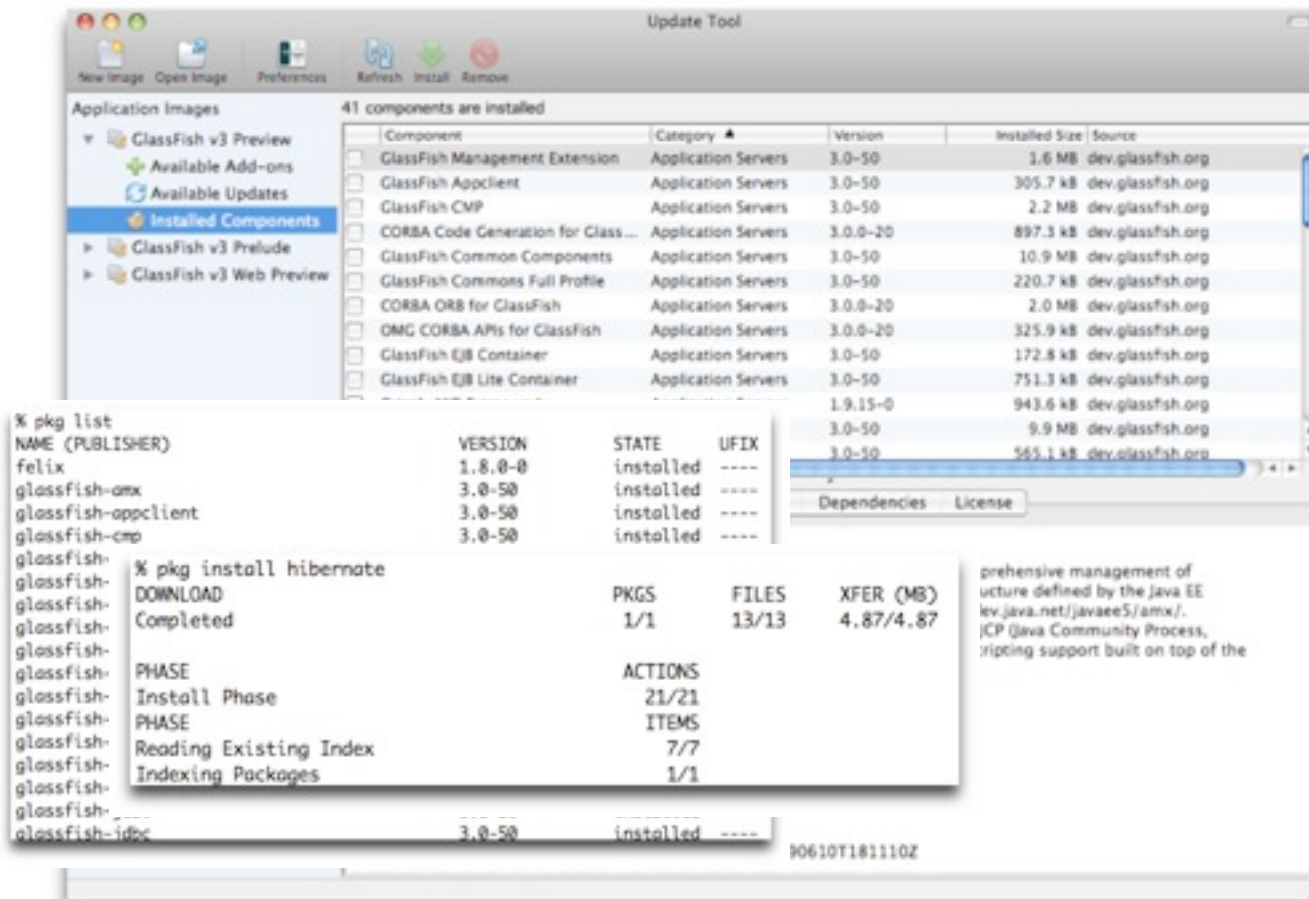
```

NAME (PUBLISHER)      VERSION      STATE      UFX
felix                 1.8.0-0     installed  ----
glassfish-omx         3.0-50      installed  ----
glassfish-applclient  3.0-50      installed  ----
glassfish-cmp         3.0-50      installed  ----
glassfish-codegen     3.0.0-20    installed  ----
glassfish-common      3.0-50      installed  ----
glassfish-common-full 3.0-50      installed  ----
glassfish-corba       3.0.0-20    installed  ----
glassfish-corba-omgapi 3.0.0-20    installed  ----
glassfish-ejb         3.0-50      installed  ----
glassfish-ejb-lite    3.0-50      installed  ----
glassfish-grizzly     1.9.15-0    installed  ----
glassfish-gui         3.0-50      installed  ----
glassfish-hk2         3.0-50      installed  ----
glassfish-jca         3.0-50      installed  ----
glassfish-jcdi        3.0-50      installed  ----
glassfish-idec        3.0-50      installed  ----
  
```

The bottom right of the terminal window shows the ID `90610T183110Z`.

The screenshot shows the GlassFish v3 Admin console interface. The top bar includes buttons for 'Home' and 'Version'. Below the bar, it says 'User: anonymous Domain: c' and 'GlassFish™ v3 Admin'. A message states 'There are 1 update(s) available'. The left sidebar shows a 'Tree' view with 'Common Tasks' and a list of tasks: 'Registration', 'GlassFish News', 'Application Server', 'Applications', 'Resources', 'JDBC', 'Configuration', 'Web Container', 'Transaction Service', 'HTTP Service', 'Virtual Servers', 'Thread Pools', 'Network Config', 'Monitoring', 'Security', 'System Properties', and 'Update Tool' (selected).

# Update Center





# Update Center

The screenshot shows the GlassFish v3 Update Tool window. The top bar includes buttons for 'New Image', 'Open Image', 'Preferences', 'Refresh', 'Install', and 'Remove'. The left sidebar shows 'Application Images' with options like 'GlassFish v3 Preview', 'Available Add-ons', 'Available Updates', and 'Installed Components'. The main area lists 41 installed components with columns for Component, Category, Version, Installed Size, and Source. Below this, a terminal window displays the output of the '% pkg list' command, showing a table of installed packages and their states. A tooltip for '% pkg Usage:' is also visible.

Component	Category	Version	Installed Size	Source
GlassFish Management Extension	Application Servers	3.0-50	1.6 MB	dev.glassfish.org
GlassFish Applclient	Application Servers	3.0-50	305.7 kB	dev.glassfish.org
GlassFish CMP	Application Servers	3.0-50	2.2 MB	dev.glassfish.org
CORBA Code Generation for Class...	Application Servers	3.0.0-20	897.3 kB	dev.glassfish.org
GlassFish Common Components	Application Servers	3.0-50	10.9 MB	dev.glassfish.org
GlassFish Commons Full Profile	Application Servers	3.0-50	220.7 kB	dev.glassfish.org
CORBA ORB for GlassFish	Application Servers	3.0.0-20	2.0 MB	dev.glassfish.org
OMG CORBA APIs for GlassFish	Application Servers	3.0.0-20	325.9 kB	dev.glassfish.org
GlassFish EJB Container	Application Servers	3.0-50	172.8 kB	dev.glassfish.org
GlassFish EJB Lite Container	Application Servers	3.0-50	751.3 kB	dev.glassfish.org

NAME (PUBLISHER)	VERSION	STATE	UFIX
felix	1.8.0-0	installed	----
glassfish-amx	3.0-50	installed	----
glassfish-applclient	3.0-50	installed	----
glassfish-cmp	3.0-50	installed	----

The screenshot shows the GlassFish v3 Admin console. The top bar includes 'Home' and 'Version' buttons. The main area displays 'User: anonymous' and 'Domain: c'. Below this, it says 'GlassFish™ v3 Admin' and 'There are 1 update(s) available'. The left sidebar shows a 'Tree' view with 'Common Tasks' and 'Configuration' sections. The 'Update Tool' is highlighted in the 'Configuration' section.



# OSGi benefits for GlassFish

- Demands and enforces stronger modularity
- Enables custom tailored App server
- Lazy loading based on usage patterns
- Open for all JVM based technologies
  - Native deployment of JRuby-on-Rails application
- Successfully maintained quick startup
- Available to GlassFish developers and users

# What's the deal with OSGi?

- GlassFish runs on top of OSGi (Felix by default)
  - Also runs unmodified on Equinox
  - GlassFish ships as 200+ bundles
  - Can run without OSGi (Static mode)
  - Can use OSGi management tools (CLI or Web)
  - GlassFish OSGi Web Admin Console
  - Can be installed on top of existing OSGi runtimes

Apache Felix Web Console - X

http://localhost:8080/osgi/system/console/bundles/

# Apache Felix Web Console Bundles

APACHE felix

Bundles Configuration Status Licenses Log Service OSGi Repository Services Shell System Information

Bundle information: 200 bundles in total, 59 bundles active, 13 active fragments, 127 bundles resolved, 1 bundles installed.

Apply Filter Filter All Reload Install/Update... Refresh Packages

Id	Name	Version	Category	Status	Actions
0	System Bundle (org.apache.felix.framework)	3.0.1		Active	
106	A Pluggable InMemory JACC Provider (org.glassfish.security.inmemory.jacc.provider)	3.1.0.b10		Resolved	
153	Admin CLI for v3 (org.glassfish.admin.cli)	3.1.0.b10		Resolved	
100	Admin Console Clustering Support Plugin (org.glassfish.admingui.console-cluster-plugin)	3.1.0.b10		Resolved	
29	Admin Console Common (org.glassfish.admingui.console-common)	3.1.0.b10		Resolved	
184	Admin Console Common Help Content (org.glassfish.admingui.console-common-help)	3.0.0.b08		Fragment	
122	Admin Console Community Edition Theme Plugin (org.glassfish.admingui.console-community-branding-plugin)	3.1.0.b10		Resolved	
53	Admin Console Connectors Plugin (org.glassfish.admingui.console-jca-plugin)	3.1.0.b10		Resolved	
193	Admin Console EJB Local (Lite) Plugin (org.glassfish.admingui.console-ejb-lite-plugin)	3.1.0.b10		Resolved	
35	Admin Console Integration (org.glassfish.admingui.console-plugin-service)	3.1.0.b10		Resolved	
68	Admin Console JDBC Plugin (org.glassfish.admingui.console-jdbc-plugin)	3.1.0.b10		Resolved	
192	Admin Console Jruby Plugin (org.glassfish.admingui.console-jruby-plugin)	3.1.0.b10		Resolved	
5	Admin Console JTS Plugin (org.glassfish.admingui.console-jts-plugin)	3.1.0.b10		Resolved	
86	Admin Console Update Center Plugin (org.glassfish.admingui.console-updatecenter-plugin)	3.1.0.b10		Resolved	
128	Admin Console Web Container Plugin (org.glassfish.admingui.console-web-plugin)	3.1.0.b10		Resolved	
67	Admin Core classes (org.glassfish.admin.core)	3.1.0.b10		Resolved	
109	Admin Utilities classes (org.glassfish.admin.util)	3.1.0.b10		Resolved	
168	Administration Configuration APIs (org.glassfish.admin.config-api)	3.1.0.b10		Active	
173	Administration Console Connectors Help Content (org.glassfish.admingui.console-jca-plugin-help)	3.0.0.b08		Fragment	
42	Administration Console EJB Lite Help Content (org.glassfish.admingui.console-ejb-lite-plugin-help)	3.0.0.b08		Fragment	
87	Administration Console Java Transaction Service Help Content (org.glassfish.admingui.console-jts-plugin-help)	3.0.0.b08		Fragment	

# Does OSGi matter to you?

- As an enterprise app developer, should you care about OSGi?
- Probably, yes
  - OSGi Enterprise 4.2 from March 2010
- Improved modularity
  - Reusable bundles
  - Dependencies are more visible
  - Better isolation / Cleaner class loading model
  - Better version control
  - Faster deployment cycle
- Observable bundle life cycle

OSGi Service Platform  
Enterprise Specification  
The OSGi Alliance

Release 4, Version 4.2  
March 2010

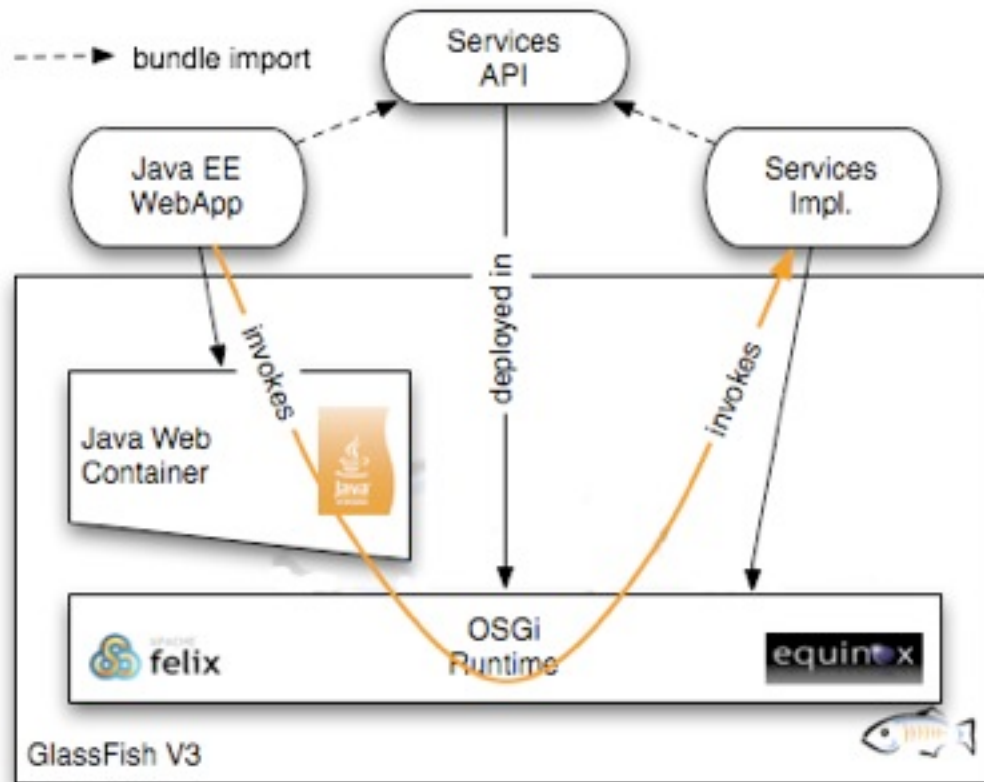


# What's the deal with OSGi?

- Any OSGi bundle will run in GlassFish
  - Drop it in `glassfish/modules{/autostart}`
  - Can also `asadmin` deploy it using `--type osgi`

# Extending GlassFish 3.x

## OSGi-style

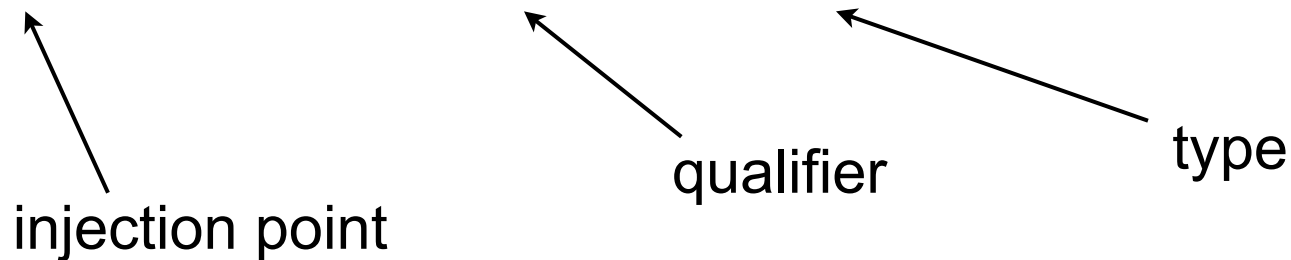


- OSGi declarative service
- Service-Component entry in the JAR Manifest
- Invoke the service from a servlet using standard @Resource injection
- Never use a GlassFish API !
- No need to choose between OSGi and Java EE

# OSGi and CDI

- While standard, @Resource is a bit arcane
  - Static binding
- Java EE 6 introduces CDI (Context & Dependency Injection) to unify and standardize injection

```
@Inject @LoggedIn User myUser;
```



- Strong typing (no strings involved), yet loose coupling
- *(no XML config either)*

# Introducing @OSGiService

```
@Inject
```

```
@OSGiService(/*dynamically bound*/ dynamic=true,  
              /*wait 10 seconds*/ waitTimeout=10*1000)
```

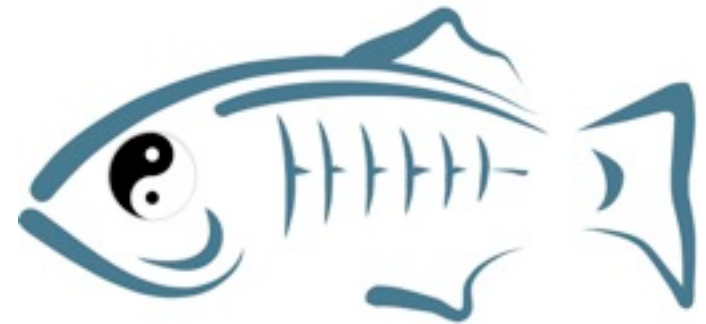
```
PaymentService myPaymentEngine;
```

- OSGiServiceExtension kicks in to discover and instantiate the service implementing the interface type.
- Offers dynamic behavior which many expect from OSGi
- (requires CDI's beans.xml)



# Introducing Hybrid Applications

- An artifact that is both :
  - A Java EE archive
  - An OSGi bundle
- Requires a runtime to fully support both standards



# GlassFish & OSGi for developers

- Single, all-in-one modular runtime
  - Assembled spontaneously with proper tooling
  - Full Java EE 6 support
- Java EE-related OSGi services & standards
  - OSGi/HTTP Service
  - OSGi/JDBC (rfc #122)
  - OSGi/JTA (rfc #98)
  - OSGi/JPA (rfc #143)
  - OSGi/JNDI (rfc #142)
  - OSGi/Web Application (rfc #66)

# Web Application Bundle (WAB)

OSGi/Web Application (rfc #66)

- WAR + OSGi bundle configuration

Manifest-Version: 1.0

Import-Package: javax.servlet.http; javax.persistence

Bundle-ClassPath: WEB-INF/classes/,  
WEB-INF/lib/entities.jar

Bundle-Version: 1.0

Bundle-ManifestVersion: 2

*Required*

**Web-ContextPath: /hello**

Bundle-SymbolicName: test.hellowab

- Lifecycle can now be driven by OSGi
- Wrapped WAR Support

webbundle:file:///myWebApp.war?Web-ContextPath=/hello

# Hello Hybrid World!

```
hellohybrid/UserAuthService.class  
hellohybrid/impl/Activator.class  
hellohybrid/impl/UserAuthServiceEJB.class  
hellohybrid/impl/RegistrationServlet.class  
hellohybrid/impl/LoginServlet.class  
META-INF/MANIFEST.MF
```

# Hello Hybrid World!

Service interface



hellohybrid/UserAuthService.class

hellohybrid/impl/Activator.class

hellohybrid/impl/UserAuthServiceEJB.class

hellohybrid/impl/RegistrationServlet.class

hellohybrid/impl/LoginServlet.class

META-INF/MANIFEST.MF

# Hello Hybrid World!

Service interface

Creates DB table  
after deployment

hellohybrid/UserAuthService.class

hellohybrid/impl/Activator.class

hellohybrid/impl/UserAuthServiceEJB.class

hellohybrid/impl/RegistrationServlet.class

hellohybrid/impl/LoginServlet.class

META-INF/MANIFEST.MF

# Hello Hybrid World!

Service interface

Creates DB table  
after deployment

Service  
implementation

hellohybrid/UserAuthService.class

hellohybrid/impl/Activator.class

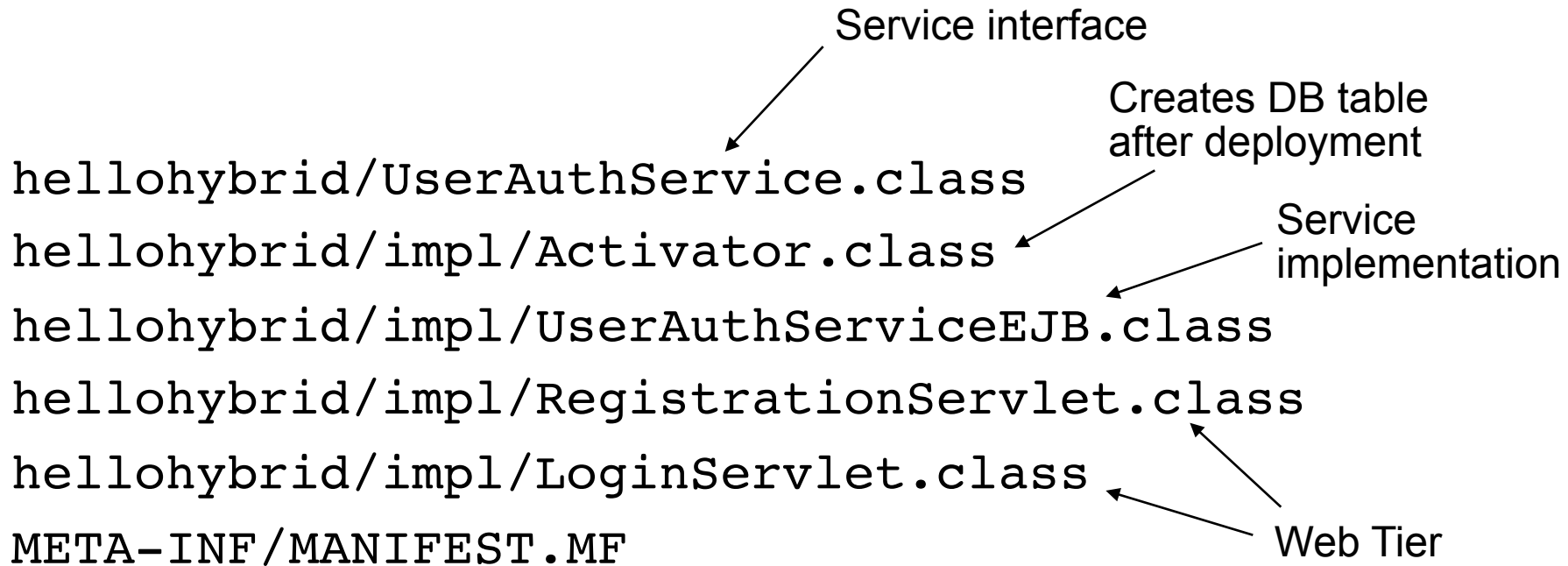
hellohybrid/impl/UserAuthServiceEJB.class

hellohybrid/impl/RegistrationServlet.class

hellohybrid/impl/LoginServlet.class

META-INF/MANIFEST.MF

# Hello Hybrid World!

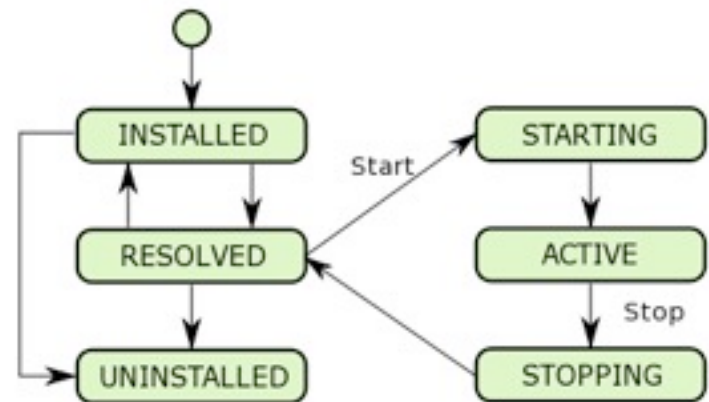




# WAB Scenario

```
telnet localhost 6666
install file://work/hybridapp.war/target/hybridapp.war
start
(web application getting deployed. Access the URL/application.)
stop
(the web app gets undeployed and URL/application are no more accessible.)
start
(it's back again)
uninstall
(the bundle gets removed from the system.)
```

- Java EE deployment on  
RESOLVED => STARTING
- Java EE un-deployment on  
STOPPING => RESOLVED

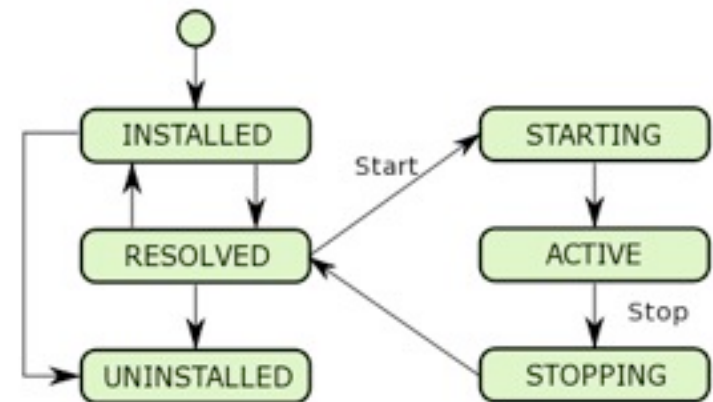


# WAB Scenario

OSGi metadata  
already present

```
telnet localhost 6666
install file://work/hybridapp.war/target/hybridapp.war
start
(web application getting deployed. Access the URL/application.)
stop
(the web app gets undeployed and URL/application are no more accessible.)
start
(it's back again)
uninstall
(the bundle gets removed from the system.)
```

- Java EE deployment on  
RESOLVED => STARTING
- Java EE un-deployment on  
STOPPING => RESOLVED



# Alternate WAB deployments

- With OSGi manifest

```
asadmin deploy --type osgi hybridapp.war
```

```
cp hybridapp.war glassfish/modules/autostart
```

```
cp hybridapp.war domains/domain1/autodeploy/bundles
```

- Plain WAR file

```
install
```

```
webbundle:file:///work/webapp/target/webapp.war?
```

```
Web-ContextPath=/hello
```

# OSGi/EJB

(GlassFish Feature)

- Allow EJBs to run within OSGi context
  - Easily discovered/accessed by other OSGi bundles
  - Can use container managed JPA
  - TX, security, persistence contexts propagated
- Automatic enlistment of EJBs as OSGi services
  - Local beans only (SFSB, SLSB, Singleton)
  - Manifest entry Export-EJB: ALL|NONE
- Can split client & service into separate bundles
  - OSGi ServiceTracker listens for EJB service availability, making deployment order irrelevant

# OSGi/JPA

- JPA in hybrid applications
- Support multiple JPA providers for OSGi enterprise applications
- Standalone persistence unit in an OSGi bundle
  - Entities + `persistence.xml`
  - Export the EMF as a service
  - Shared PU means shared 2nd level cache
- Injectable with `@OSGiService`

# Summary

- GlassFish is modular (OSGi) & extensible (HK2)
- OSGi Declarative Services to extend GlassFish
  - No OSGi API, no GlassFish API, Integration Nirvana!
- CDI injection for dynamic OSGi services
- Hybrid applications bring OSGi benefits to JavaEE (and vice-versa)
  - Reusable bundles
  - Independent life-cycles
  - Standard and existing application model
- GlassFish covers your Java EE and OSGi needs

# Resources

- Sahoo's blog: <http://weblogs.java.net/blog/ss141213>
- Jerome's blog : <http://blogs.sun.com/dochez>
- <http://wikis.sun.com/display/GlassFish/OsgiDashboard>
- <http://glassfish.org>



Main sponsor



alexis.moussine-pouchkine@oracle.com  
@alexismp

<http://blogs.sun.com/alexismp>  
<http://blogs.sun.com/theaquarium>







Main sponsor



# HTML5 WebSockets - the Web Communication revolution, making the impossible, possible

Brad Drysdale

Picasso

Matejko + Canaletto

Malczewski + Chelmonski

State of Scala

Venkat Subramaniam

Don't code - create  
software!

Paweł Lipiński

Java Boilerplate Busters

Hamlet D'Arcy