SSI SCHAEFER

Noch einmal OSGi
Salomon Automation GmbH, Austria

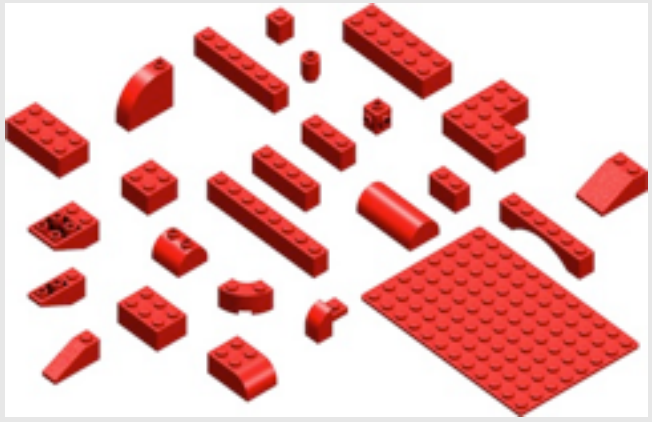
Istvan Benedek <istvan.benedek@salomon.at>

Excellence in Motion

What is OSGi?

"OSGi technology provides a service-oriented, component-based environment for developers and offers standardized ways to manage the software lifecycle."[1]

"Modularity is a core concept necessary for the future of Java application architecture"[2]



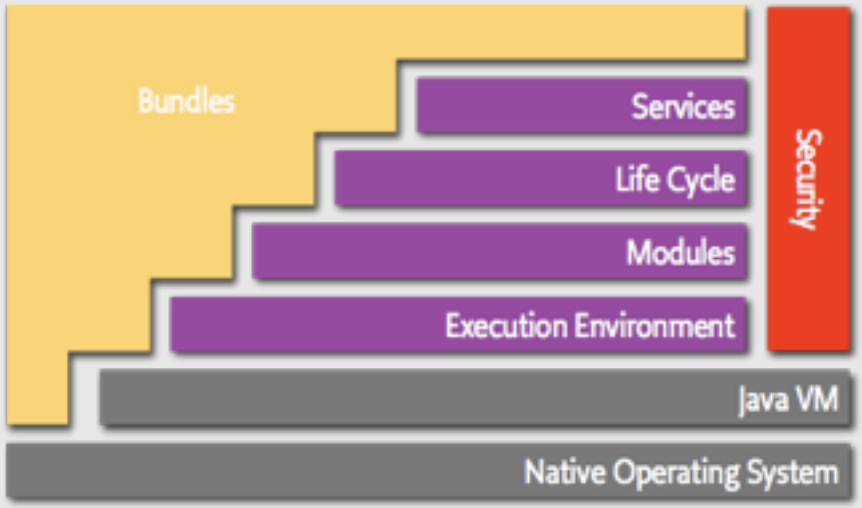
Picture 1

Picture 1: http://dev.legoimages.com, http://dev.legoimages.com/images/duplo/lego-bricks-high-resolution.jpg

[1] OSGi Module System, http://www.osgi.org
[2] Knoernschild, Kirk (2012). Java Application Architecture, Modularity Patterns with Examples Using OSGi,Prentice Hall, pp. 317.

Architecture

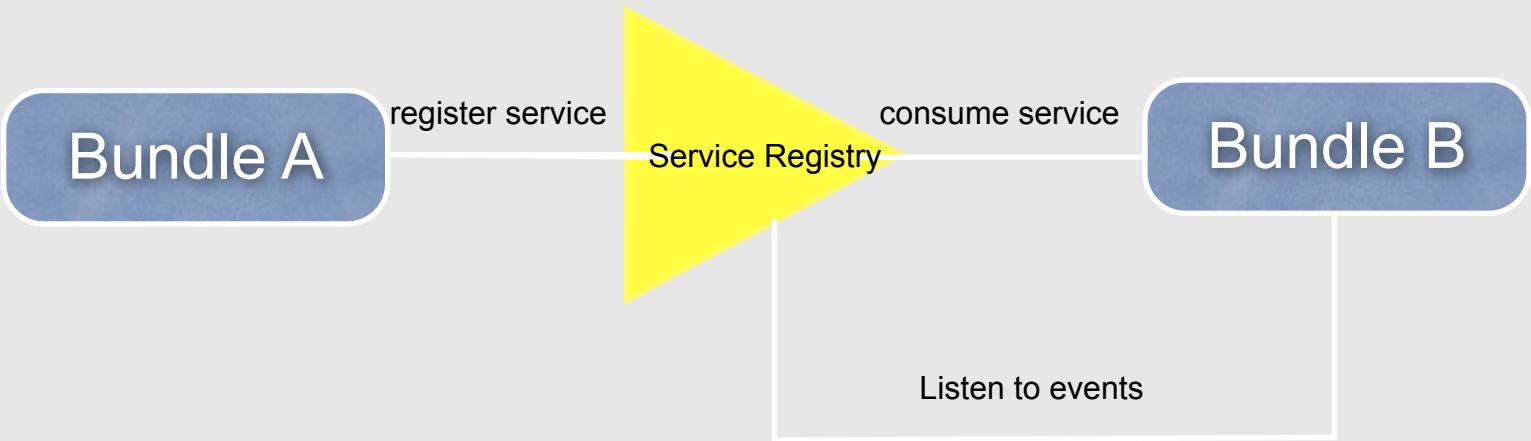
- **Bundles** - OSGi components which are created by the developers.
- **Services** - The bundles can connect to each other through the services layer, which provides publish-find-bind model for POJOs.
- **Life-Cycle** - API to perform the following actions with bundles: install, start, stop, update, uninstall.
- **Modules** - encapsulation, import and export code.
- **Security** - orthogonal layer that handles the security aspects.
- **Execution Environment** - methods and classes which are available in a specific platform.



Picture 2

Picture 2: http://www.osgi.org/Technology/WhatIsOSGi

Service Registry



Picture 3

Picture 3: based on http://www.osgi.org/Technology/WhatIsOSGi

Bundle=Plugin≠Batyu

- Bundle contains java classes, resources and a bundle descriptor called manifest.
- Every bundle is a jar, but not every jar is a bundle
- bundle is a self-describing collection of files
- META-INF/MANIFEST.MF
- Exception: Exploded bundle (not so trivial to use it)
- Bundle has own encapsulation
- 1 classloader/bundle @runtime => we have many class instances of the same class

The MANIFEST.MF

```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: Popup Plug-in
Bundle-SymbolicName: com.example.myosgi;
singleton:=true
Bundle-Version: 1.0.0
Bundle-Activator:
com.example.myosgi.Activator
Require-Bundle: org.eclipse.ui,
 org.eclipse.core.runtime
Bundle-ActivationPolicy: lazy
Bundle-RequiredExecutionEnvironment:
JavaSE-1.6
```

Table 1

Identifier	Description
Bundle-Name	Short descriptive text of the bundle.
Bundle-SymbolicName	The unique identifier for this bundle.
Bundle-Version	Defines the bundle version and must be incremented if a new version of the bundle is
Bundle-Activator	Optional, defines an Activator. This class will be notified whenever the bundle is started or
Bundle-RequiredExecutionEnvironment	Specify which Java version is required to run the bundle. If this requirement is not fulfilled then the OSGi runtime does not load the
Bundle-ActivationPolicy	Setting this to *lazy* will tell the OSGi runtime that this plug-in should only be activated if one of its components, i.e. classes and interfaces are used by other bundles. If not set, the OSGi runtime will activate this bundle by default.

Table 2

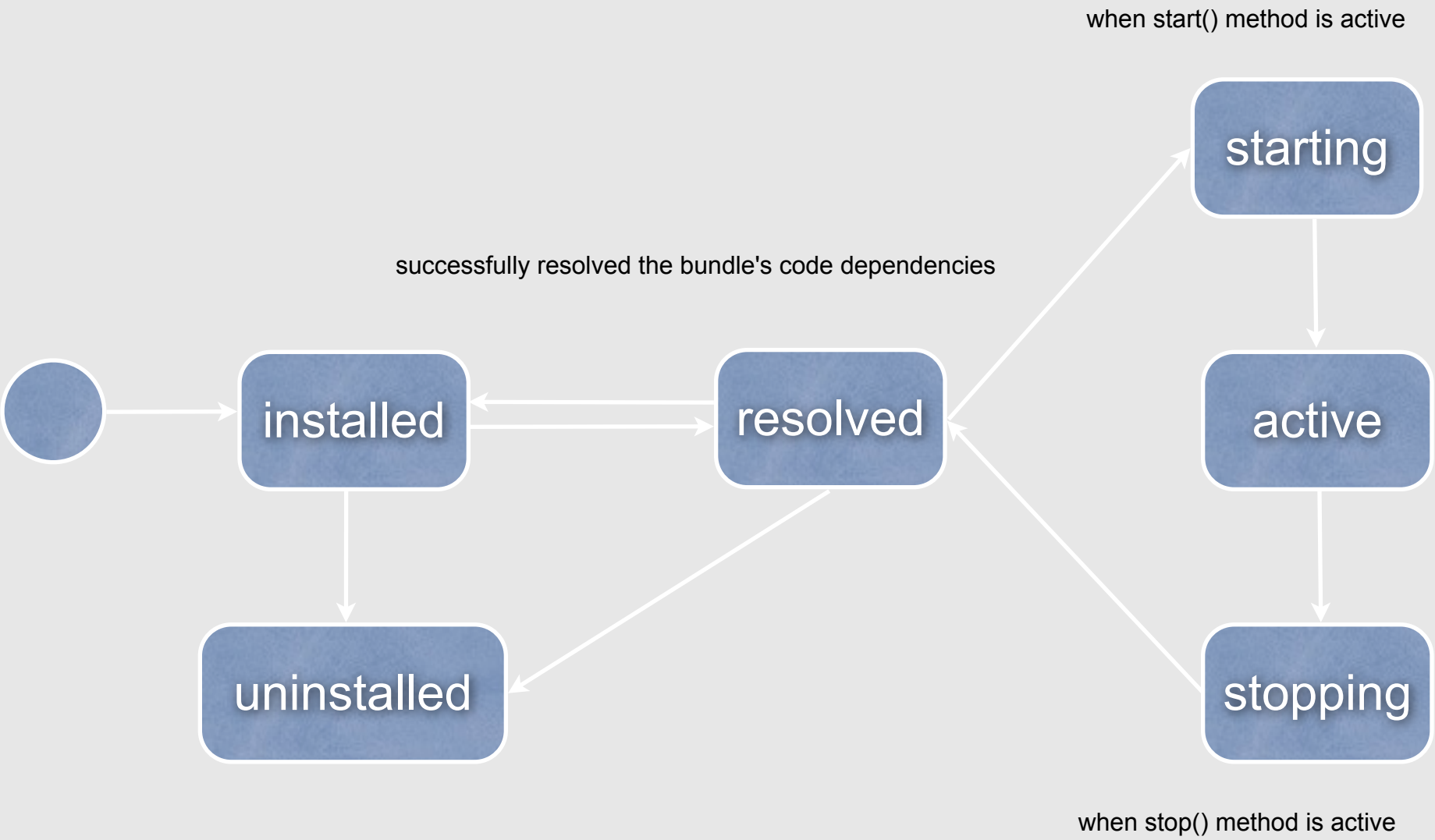
OSGi implementation specific things

x-internal, x-friends, Eclipse-LazyStart[3]

Table 1, Table 2: Vogel, Lars. "OSGi Modularity - Tutorial" Article. 5 August 2008. <http://www.vogella.com/articles/OSGi/article.html>

[3] http://help.eclipse.org/juno/topic/org.eclipse.platform.doc.isv/reference/misc/bundle_manifest.html

Bundle life-cycle



Picture 4

Picture 4 based on: OSGi "Life-cycle" chapter. 13 November 2012. <http://en.wikipedia.org/wiki/OSGi>

OSGi implementations

Knopflerfish 3 OSGi R4 v4.2 (small, +GUI application to handle bundles)

Apache Felix Framework [GlassFish, JBoss AS 7]

ProSyst Software mBedded Server 7

Hitachi Solutions SuperJ Engine Framework V4

Equinox 3.2 [Eclipse] OSGi R4 compliant, Reference implementation

Further information: http://www.osgi.org/Certification/Certified

History

1998 November - OSGi Alliance was founded
2003 October - OSGi R3
2004 Eclipse 3.0 based on OSGi
2005 October - OSGi R4
2007 May - OSGi R4.1
2009 September - OSGi R4.2 (framework launching, remote service, blueprint service, Enterprise Specification: web applications, JPA service, JNDI Service, JDBC Service)
2011 April - OSGi R4.3
2012 OSGi 5

Present & Future

OSGi is definitely not dead, OSGi 5 implementations are coming soon...

Most of the new Java EE app servers built top of OSGi, though java 9 and Jigsaw can kill it...

201? - funeral of OSGi

Further information: http://blog.osgi.org/2008/12/project-jigsaw.html; http://en.wikipedia.org/wiki/OSGi

Terminology

- **EJB** - Enterprise Java Bean
- **Modular EJB** - An EJB running in OSGi
- **EJB Bundle** - An OSGi bundle packaging Modular EJBs
- **WAB** - Web application bundle
- **CDI** - context and Dependency Injection (@EJB, @Resource etc.)
- **IoC** - Inversion of Control
- **bi-directiona**l interaction between OSGi and Java EE component
- It is possible to start a web server or even a GlassFish server to start from a bundle
- **Hybrid Application =** OSGi + JAVA EE

Benefits

- Reduce complexity, well defined interfaces
- Reduce maintenance cost
- Reusable component
- Widely used, there are many implementations, tools which support OSGI
- Easy deployment, dynamic updates
- Versioning
- Standard and Enterprise ready
- Rapid development
- Standard technologies (JTA, JMS, JAAS, EJB, JPA, JNDI, JAXB, JAX-WS, JAX-RS, JAXP, JMX, JSF, IoC, CDI, JCA, JavaMail)[4]
- Scalable, extendable, reliable
- Widely used, there are many JAVA EE 6 server implementations

[4]Further information:http://en.wikipedia.org/wiki/Java_Platform,_Enterprise_Edition

Case study

Story

Given Loandry, a Financial institution which provides different kind of loans for their clients.

Manager Rob would like to get a modular, scalable application, which can work together with the legacy systems. The new application shall be able to solve the business needs for the next 5 years.

Manager Rob persuades the sponsors to approve a prototype project and assigns the task to Coder Bill...

Applied technologies

JAVA 7

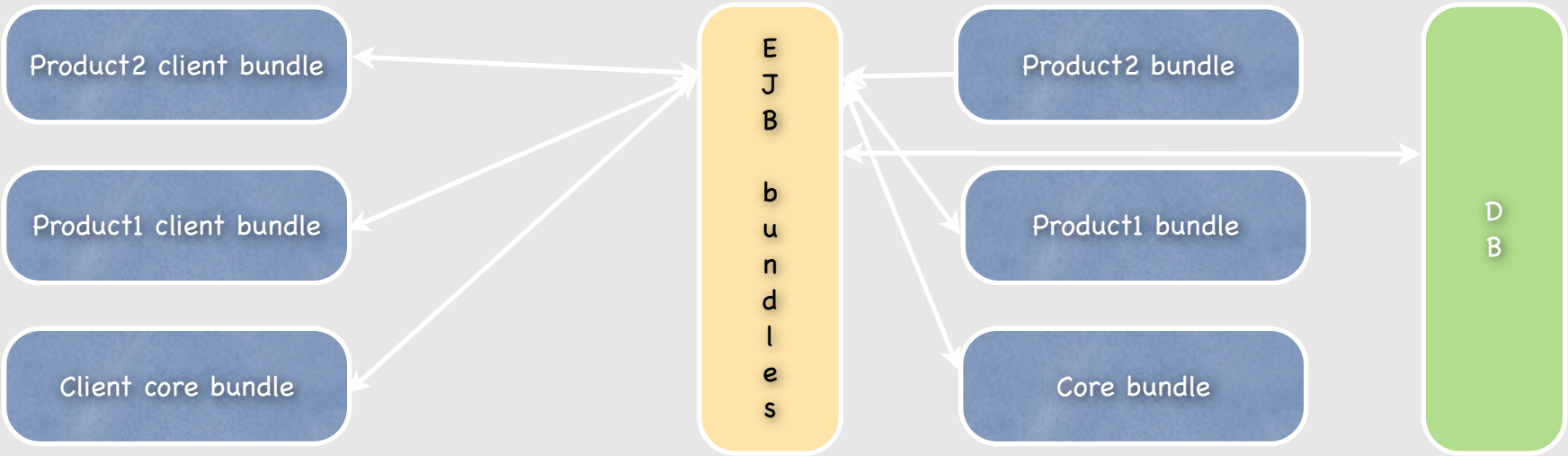
Glassfish 3.1.2.2 JAVA EE 6 certified AS (Other EE 6 cerfitfied Application servers: Oracle Weblogic Server, JBoss AS 7.1, JBoss Enterprise Application Platform 6.0, IBM Websphere AS 8, IBM Websphere AS CE 3.0, Apache Geronimo 3.0, Fujitsu Interstage AS 1, TmaxSoft JEUS 7)[5]

MySQL 5.1.51

Eclipse 4.2

[5] Further information:http://en.wikipedia.org/wiki/Java_Platform,_Enterprise_Edition

DEMO


Product2 client bundle
Product1 client bundle
Client core bundle
EJB bundles
Product2 bundle
Product1 bundle
Core bundle
DB

Reference list

- McAffer, Jeff; VanderLei, Paul; Archer, Simon (February 2010). OSGi and Equinox: Creating Highly Modular Java Systems (1st ed.). Addison-Wesley Professional.
- ReSTful OSGi Web Applications - Tutorial, http://www.eclipsecon.org/2008/sub/attachments/ReSTful_OSGi_Web_Applications_Tutorial.pdf
- OSGi Modularity - Tutorial, http://www.vogella.com/articles/OSGi/article.html
- OSGi: A Proven Approach for Developing Modular Applications, http://theserverside.com/feature/OSGi-is-the-framework-for-all-modular-framework
- Knopflerfish OSGi Tutorial v2, Sven Haiges, Erik Wistrand, http://www.knopflerfish.org/releases/current/docs/tutorials/kf_osgi_tutorial.pdf
- OSGi Application Development using Glassfish Server Version 1.5, Sanjeeb Sahoo, http://glassfish.java.net/public/GF-OSGi-Features.pdf
- Richard S. Hall;Karl Pauls; Stuart McCulloch; David Savage (2011). OSGi in Action. Manning publication.
- Knoernschild, Kirk (2012). Java Application Architecture, Modularity Patterns with Examples Using OSGi,Prentice Hall
- google://osgi+java+EE

SSI Schaefer –
we are there for you …