

Software jako Lego™

- úvod do komponent a OSGi

Přemek Brada, KIV ZČU
TSI 22.9.2010

Agenda

- > Software skládaný z komponent
 - > proč
 - > jak
- > OSGi
 - > co je to zač
 - > komponenta v OSGi
 - > classloader vazby – module layer
 - > dynamické vazby – service layer
- > Výzkum s OSGi na KIV
 - > co proč a jak děláme

Software skládaný z komponent

Proč komponenty

> Problémy s OOP

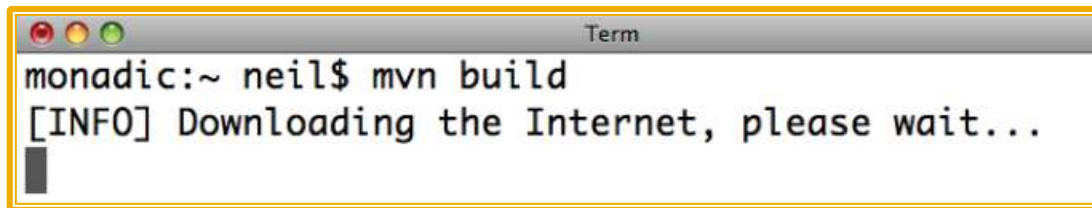
- > granularita
- > závislosti (coupling)
- > reuse?

> Common problems in Java

- > JAR HELL (no versioning)
- > the Classpath
- > class visibility
- > hot deployment

"Object Oriented technology was going to change the world ... we would have all these objects in our library and building a new system would be a snap. Just get a few classes, bunch them together ... and voila!"

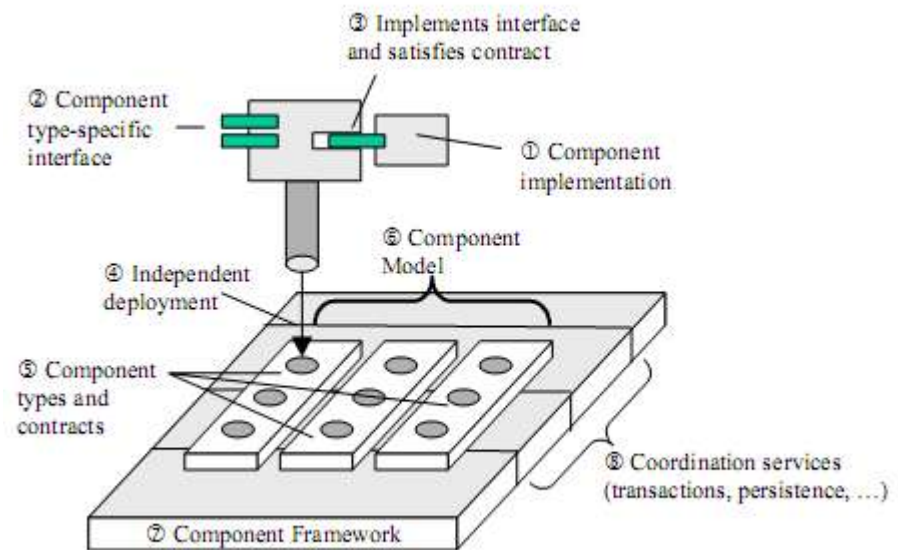
Peter Kriens, <http://www.aqute.biz/Blog/2006-04-29>

A screenshot of a terminal window titled "Term". The prompt is "monadic:~ neil\$". The command entered is "mvn build". The output is "[INFO] Downloading the Internet, please wait..." followed by a cursor.

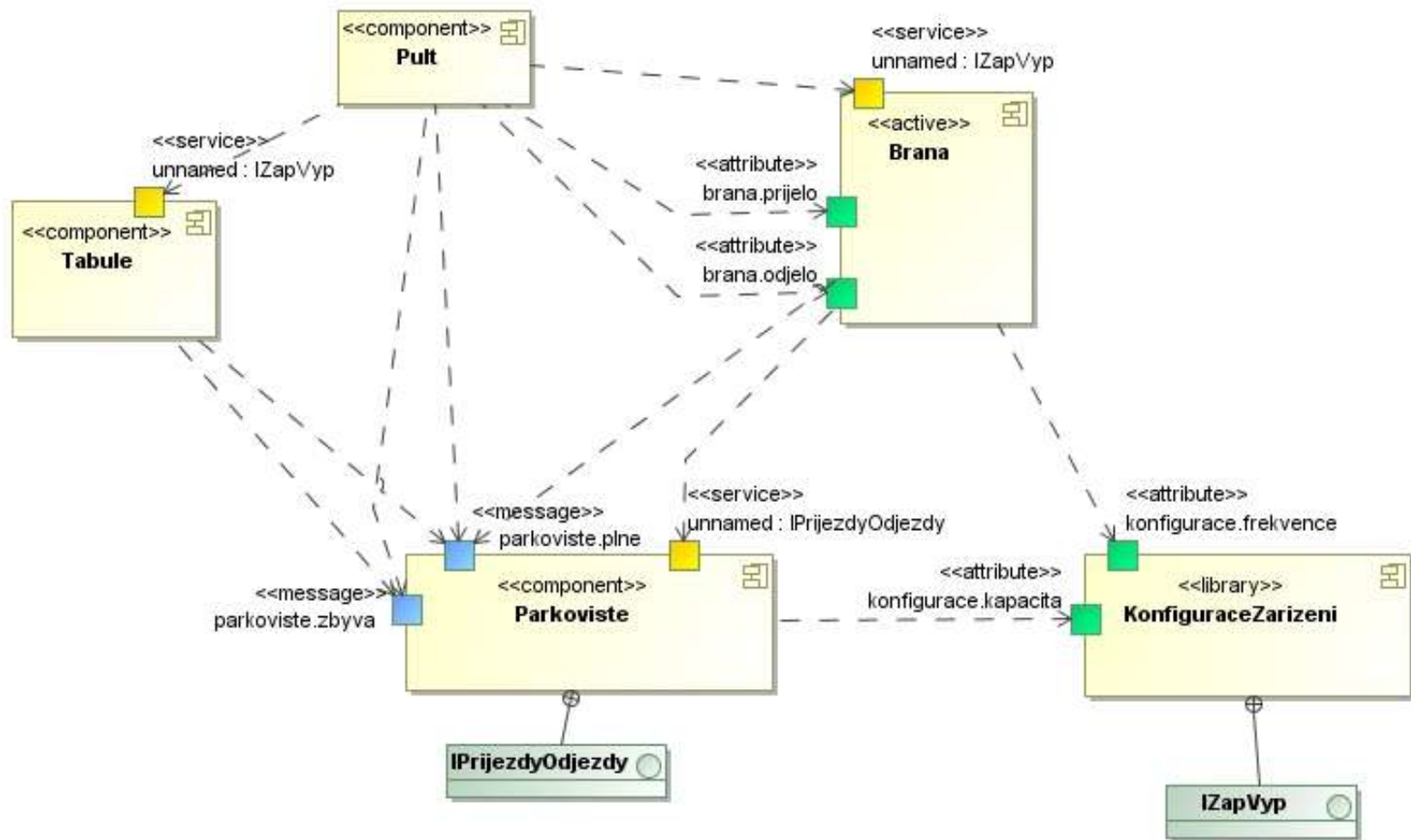
```
monadic:~ neil$ mvn build
[INFO] Downloading the Internet, please wait...
```

Jak, čili co je to komponenta

- > „Těžší“ kus software, **samostatně distribuovaný**
- > **Self-contained and declare their dependencies**
 - > declare interface
 - > hide internals
- > Vlastnosti definuje **komponentový model**
 - > žije v **kontejneru**
 - > podle **životního cyklu**



Příklad (model)



Výsledek

- > Reuse!
- > Škálovatelnost!
- > Skládání aplikací!
- > Přehlednost!

- > Obchodovatelnost!
- > Bezpečnost!

- > **Silver bullet!**



OSGi

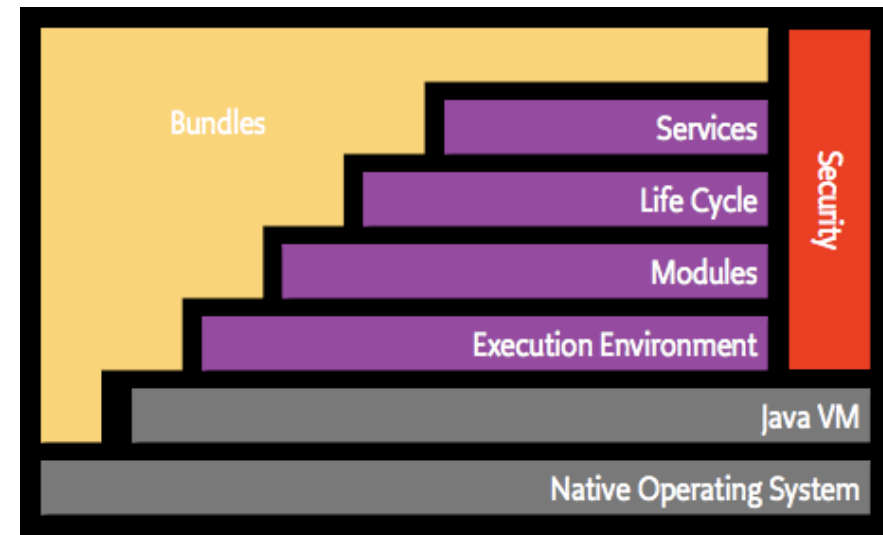
Co je zač OSGi



- > Průmyslový standard pro component-based vývoj
 - > původně home appliances, dnes až do enterprise scale
 - > Java, .jar, classloader
 - > kontejner („framework“) komerční i FOSS – Equinox, Felix, ...
 - > core (komponentový model)
+ compendium (std služby)

- > Konsorcium

- > standardizace (R4.2)
 - > popularizace



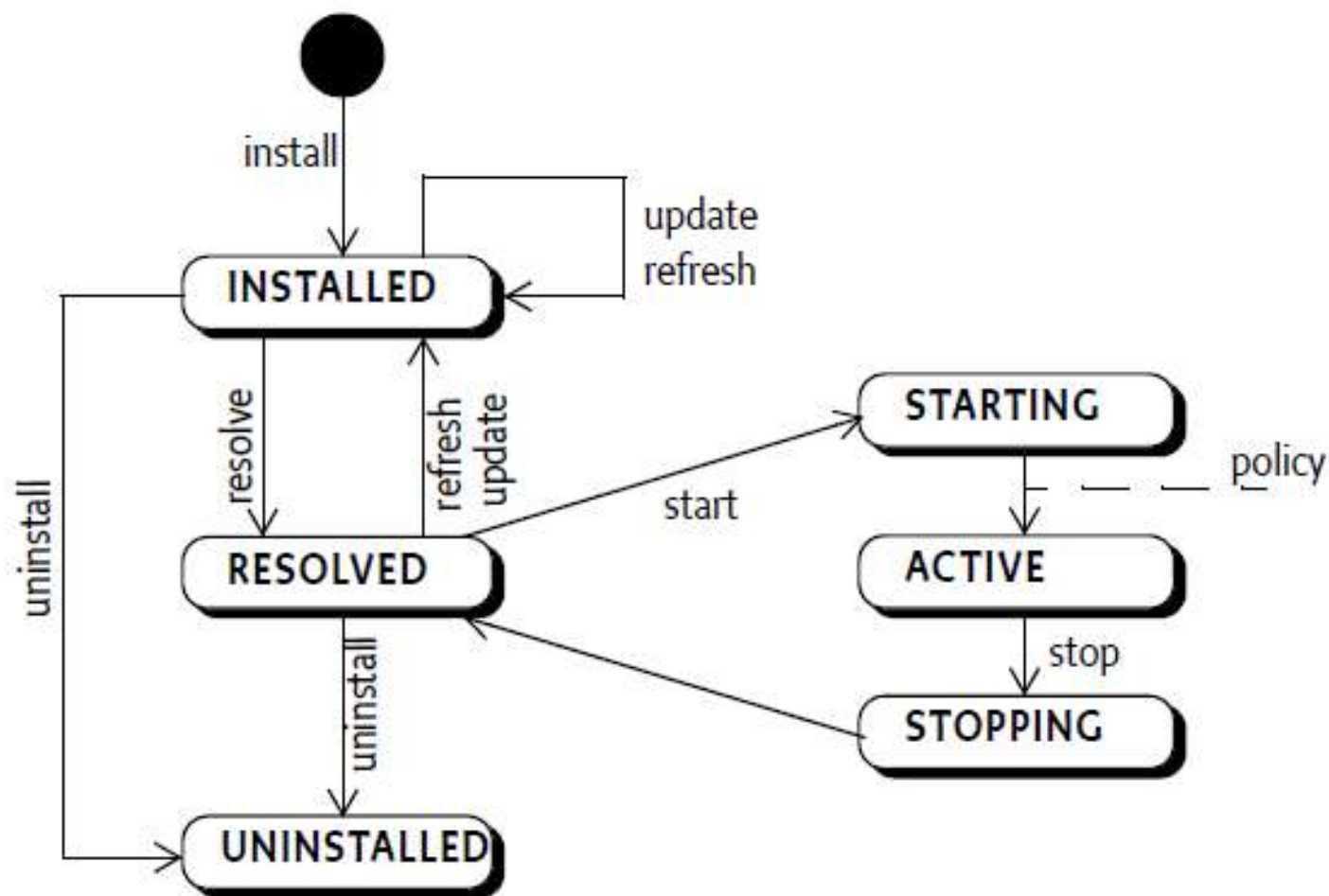
Bundle: komponenta v OSGi

- > JAR – *.class a cokoli dalšího
- > Manifest – deklarace
 - > exportované a importované prvky
 - > verze
 - > ...
- > Aktivátor, vlastní služby

```
Manifest-Version: 1.0
Bundle-Name: SimpleGateBundle
Bundle-Activator: SimpleGateMain
Bundle-SymbolicName: cz.zcu.kiv.gate
Bundle-Version: 1.0.2
Export-Package: cz.zcu.kiv.osgi.Brana
Import-Package: org.osgi.framework;
version="1.3.1"
```

- > Bundle context – reprezentace frameworku
 - > řízení životního cyklu, reakce na události
 - > vazby na okolí (properties, services, ...)

Životní cyklus bundle



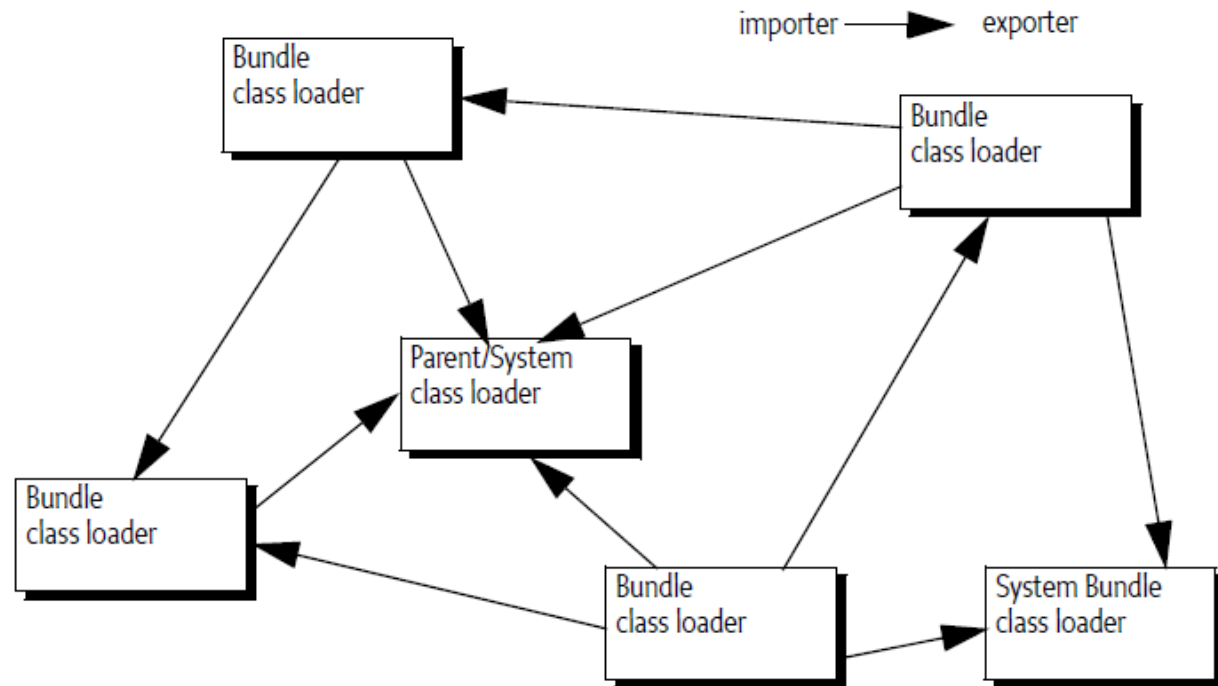
Statické vazby – module layer

> Problém: jak zajistit konzistenci class space

- > nemáme monolit
- > více exportérů
- > více verzí téhož

> Řešení

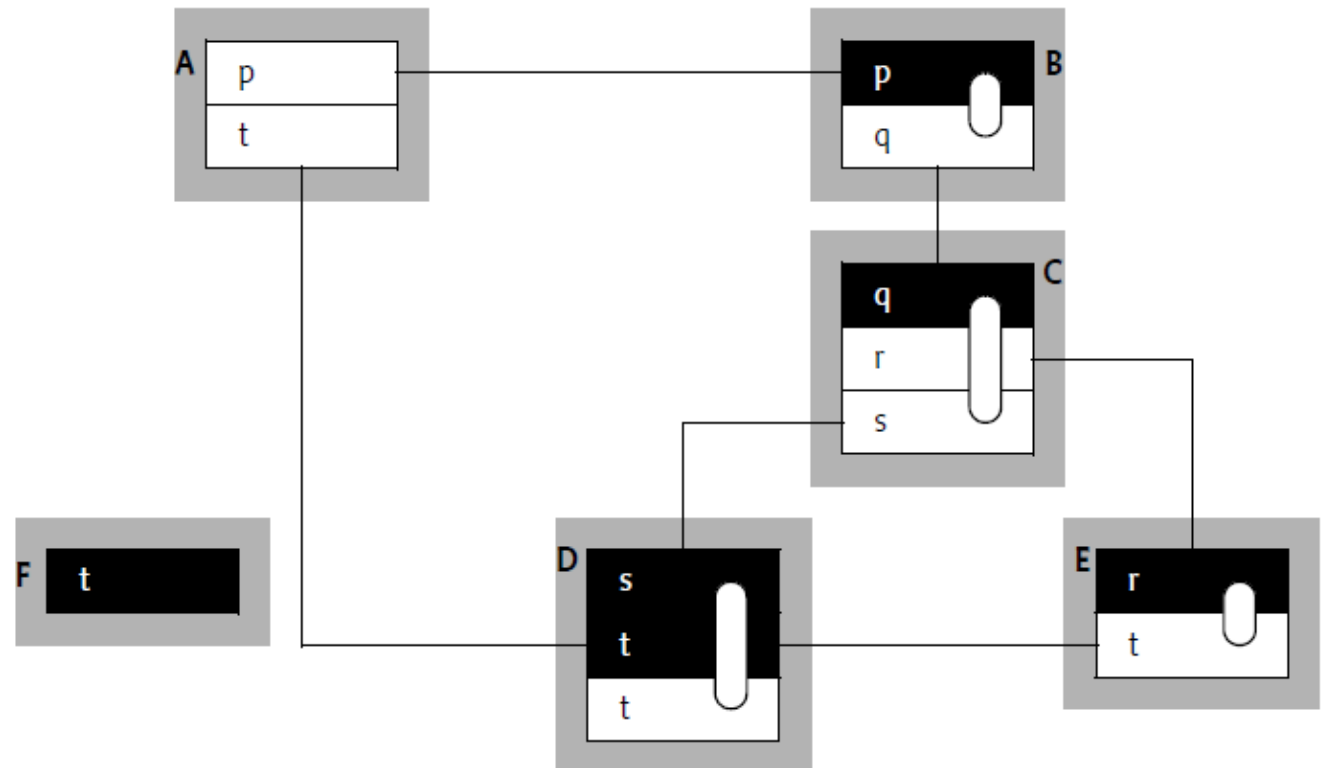
- > bundle má vlastní classloader
- > ten deleguje na exportéry
- > framework vytváří wires mezi bundly



Module layer – řešení závislostí

> Package name, version, attributes (constraints)

> Exporter sharing



Dynamické vazby – service layer

> Problém

- > chceme oddělit rozhraní (veřejné) od implementace (skrytá)
- > chceme dynamický vývoj/život aplikace

> Řešení

- > bundle poskytuje (registruje) a/nebo potřebuje (vyhledává) služby
- > různé implementace
 - > plain (OSGi core)
 - > Declarative Services
 - > Spring DM

Dynamické vazby: plain OSGi Core

- > „Dependency lookup“ model
- > Bundle (jeho třídy)
 - > implementuje service handler, registruje službu
 - > vyhledává službu, volá její metody
- > Framework
 - > obsluhuje registr služeb
 - > rozesílá události
- > Reakce na události
 - > služby mohou dynamicky přicházet / odcházet

Dynamické vazby: Declarative services

- > „Dependency injection“ model
- > Bundle: XML deklarace služeb poskytovaných / žádaných
- > Framework: instancuje a řeší závislosti

```
<scr:component xmlns:scr="http://www.osgi.org/xmlns/scr/v1.1.0"
  enabled="true" immediate="true" name="ParkovisteDSPult">
  <implementation class="cz.zcu.kiv.cosi.parkoviste.pult.Pult"/>
  <service>
    <provide interface="org.osgi.service.event.EventHandler"/>
  </service>
  <property name="event.topics" value="parkoviste/*"/>
  <reference name="IPrijeloOdjelo" bind="setStatistika" cardinality="1..1"
    interface="cz.zcu.kiv.cosi.parkoviste.brana.IPrijeloOdjelo" policy="dynamic"/>
</scr:component>
```

Výzkum s OSGi na KIV

Co, proč a jak děláme

- > Component Substitutability
- > Kontroly souladu typů
 - > exporter-importer wiring
 - > bundle update (enhanced resolver) – 1:1, kontextové
- > Verzování
 - > automatické nastavení správných id verzí
 - > bundle, package
- > Netriviální analýza bytecode, porovnávací algoritmy

Výsledky



JAZZON09

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGIES
JUNE 22–25, 2009 ZURICH



Conference Program Registration Contact



OSGi Community Event

The OSGi Community Event 2010, Sept. 29-30 in London, is centered on real-world implementations and proven benefits of OSGi standards-based architectures. The event, "The Value of OSGi: Here and Now," is held in conjunction with JAX London 2010 Autumn Edition, and the OSGi DevCon program Sept. 29 is aligned with JAX London's program.

osgi.kiv.zcu.cz/obvs/index.html

OSGi Bundle Versioning Service

This website provides a service to version [OSGi bundles](#) according to the [Semantic Versioning](#) rules. [Details are available](#) for the curious.

How to use

Submit the JAR files of a base bundle and its sequel. The service will return (for download) the sequel bundle with versioning headers in the MANIFEST file set correctly with respect to the base bundle and the changes between it and the sequel. The name of the bundle will remain as submitted.

Older version Soubor nevybrán

Newer version Soubor nevybrán

35th Euromicro Conference

Software Engineering and Advanced Applications (SEAA)
Patras, Greece, August 27-29, 2009



 Schedule

Zdroje informací

- > <http://osgi.org/>
- > <http://www.slideshare.net/> search „osgi“
- > <http://www.kiv.zcu.cz/research/groups/dss/projects/sacco.html>
- > <http://wiki.kiv.zcu.cz/UvodDoKomponent/OSGi>