



JavaPolis 2005

Mats Ekhammar

JAVAPOLIS 2005

The European Java Climax of the Year!

Antwerpen

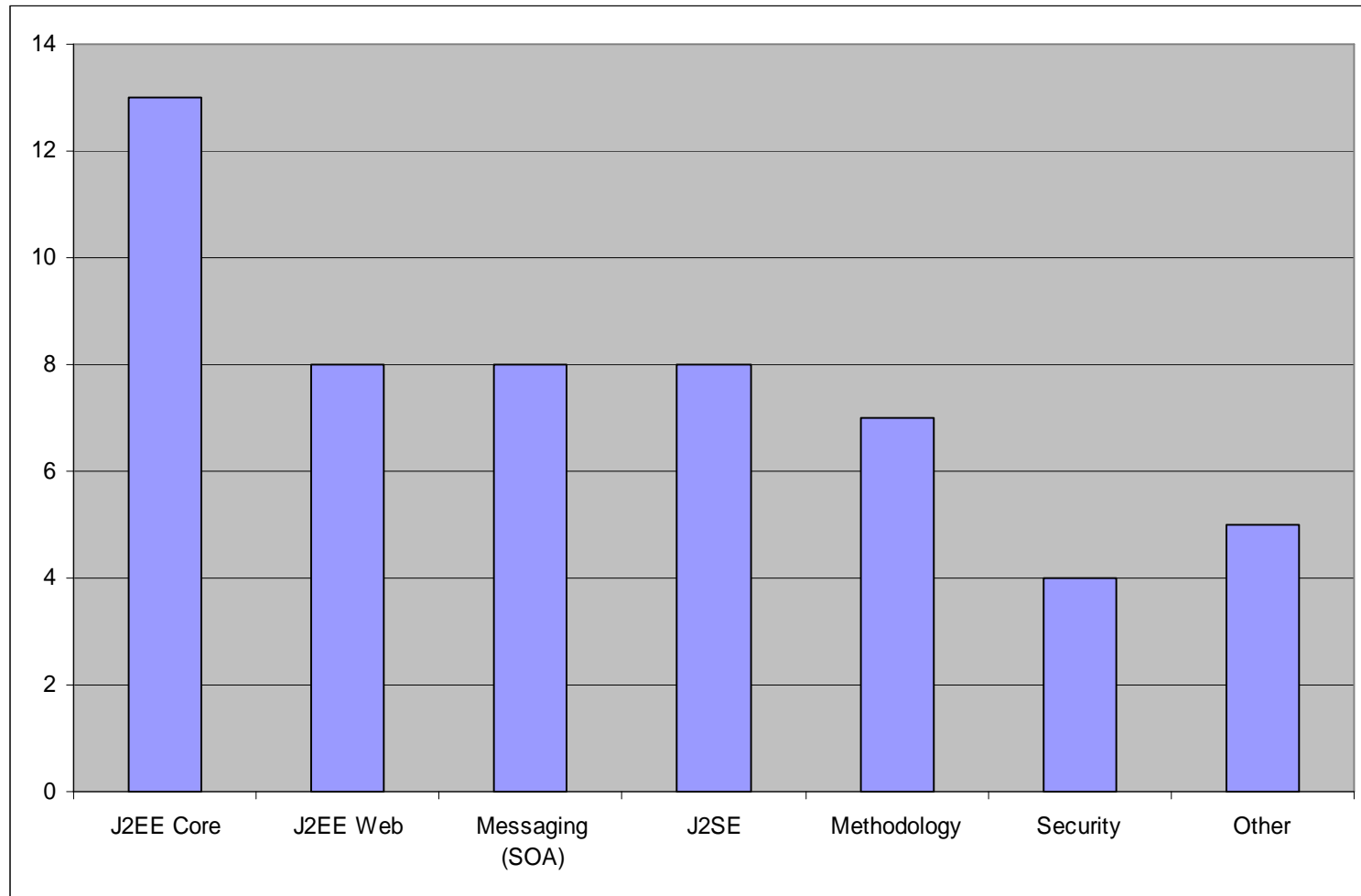


Belgien

What is it?

- Largest Java gathering in Europe!
- Arranged by Belgian Java User Group (BeJUG) since 2002.
- Cheap! 300 Euro
- 2300 attendees this year
- Held in a cinema complex!
- 5 days – University, Conference

Topics Overview



Interesting Topics!

- Java EE 5
- GlassFish
- JSF/Ajax
- Shale
- Mule
- Maven
- Spring

See: <http://wiki.javapolis.com/confluence/display/JP05/Home>

Java Puzzle

```
public class AnimalFarm {
    public static void main(String[] args) {
        final String pig = "length: 10";
        final String dog = "length: " + pig.length();
        System.out.println("Animals are equal:" + pig == dog);
    }
}
```

What does it print?

1. Animals are equal: true
2. Animals are equal: false
3. None of the above

Java Puzzle

The + operator binds tighter than ==

```
System.out.println("Animals are equal: " + pig == dog);
```

will be interpreted as

```
System.out.println( ("Animals are equal: " + pig) == dog);
```

Prints: false

Java EE 5

Java EE 5 Major Features

- Significant Ease of Development Enhancements
- New POJO-based Java Persistence API
- Simplified Web services support
- Greatly simplified EJB development
- Drag and drop Web application development with JavaServer Faces

Glassfish

Open Source Development Project that builds a Java EE 5 app server
(Sun, Oracle + community)

- Based on code from
 - Sun Java System Application Server PE 8 (J2EE 1.4)
 - Toplink persistence from Oracle
- When released
 - Sun Java System Application Server PE 9.0
 - Java EE 5 SDK
- Sun Application stack
Platform Edition -> Standard Edition -> Enterprise Edition
(compare IBM's Geronimo -> WebSphere Application Server ...)

Source code: <http://glassfish.dev.java.net>

Samples: <https://blueprints.dev.java.net/>

JSF 1.2

- JavaServer™ Faces (JSF) 1.2
 - A server side user interface (UI) component framework for web applications.
 - Not quite a new feature release but more then a maintenace release.
 - Unified JSF and JSP expression language
 - XML Schema instead of DTD
 - See: <http://www.jcp.org/en/jsr/detail?id=252>
- Development Tools
 - Sun Java Studio Creator
 - Sun Java Studio Enterprise

Struts Project

The Apache Struts project is comprised of two distinct frameworks:

- The **Struts Action Framework**, the original request-based framework
- The **Struts Shale Framework**, the component-based framework based on JavaServer Faces.

Shale

- Major Features
 - View Controller, 1:1 relationship between JSP page and JavaBean class (backing bean)
 - Dialog Manager, heavily inspired by Spring Webflow
 - Application Manager, application level controller
 - Spring Integration
 - Reusable Views
 - Test Framework

See: <http://struts.apache.org/struts-shale/index.html>

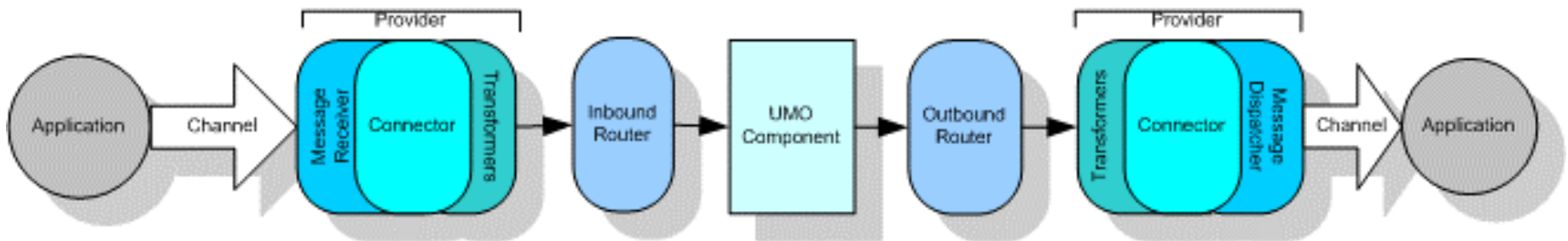
SOA

- Much talk about SOA like
 - JBI
 - WS-BPEL 2.0
 - Pragmatic SOA
 - ESB
 - SOA blueprints
 - ...

But the best show of actual doing something in the SOA area was a MULE demonstration!

MULE

Mule's ultimate goal is to provide a unified method of interacting with data from disparate sources without encumbering the developer with the details about how the data is sent or received or the protocols involved.



- UMO component is your client code that actually does something with the events received.
- Mule makes Enterprise level service architectures possible for smaller projects.

See: <http://mule.codehaus.org>

Maven 2.0

Maven is really a process of applying patterns to a build infrastructure in order to provide a coherent view of software projects.

- *Faster and smaller* - Core no longer uses Ant, Jelly or Xerces.
- *Defined build lifecycle* - Well defined phases.
- *Built-in multiple project handling*
- *No more properties files* - All configured from pom.xml.
- *No more maven.xml* - Plugins easier to build and integrate, and the only way to script your builds.
- *No more Jelly* - Plugins are primarily written in Java.

See: <http://maven.apache.org>

Spring 2.0

- XML Configuration enhancements
 - `<bean id="dataSource" class="...JndiObjectFactoryBean">`
 - `<property name="jndiName" value="jdbc/StockData"/>`
 - becomes**
 - `<jndi:lookup id="dataSource" jndiName="jdbc/StockData"/>`
- AOP enhancements
 - Closer AspectJ integration
- Scoped Beans
 - Transparent scoping for Spring beans
- Message-driven POJOs
 - “Closes off one of the remaining corner cases justifying EJB usage”

See: <http://www.springframework.org>

JavaPolis Summary

- Delivers normal conference output.
- OK mix of speakers.
- All major Java topics are addressed.
- There are still some "aha moments" out there!
- Cheap conference fee and easy travelling.