

Discussion of Paper: Dealing with Change: Components versus Services

Stefan Appel, Tobias Freudenreich, Sebastian Frischbier

A. Buchmann

Winter Term 2014/2015



TECHNISCHE
UNIVERSITÄT
DARMSTADT



DVS



**CHANGE
AHEAD**

Submitting Answers

1-2 pages means 1-2 pages, not 6.

- exceptions for those with handwriting
- no exceptions in the future
- use A4 paper
- try to use at least a 10pt font
- do not use paper clips

student id means matriculation number, not your login

Plagiarism

- you have been warned
- people I caught are excluded from the bonus
- people may notify me about plagiarism to get 0 points
 - but may still get the bonus

Definitions:

- Service: „Any act or performance that one party can offer to another that is essentially intangible and does not result in the ownership of anything.“
- Component:
 - prefabricated software part
 - encapsulate details of a set of objects

Case Study: OSGi

Wikipedia: The OSGi framework is an open, modular and scalable “Service Delivery Platform” based on Java. It is a component model with component registry (“service registry”). “Service” in the OSGi context hardly means anything more than a common interface of a component.

Creating a Bundle in OSGi*

Write a bunch of classes for some kind of functionality

Implement BundleActivator:

```
package example.helloworld;

public class Activator implements BundleActivator {
    public void start(BundleContext context) throws Exception {
        System.out.println("Hello world");
    }
    public void stop(BundleContext context) throws Exception {
        System.out.println("Goodbye World");
    }
}
```

Write Manifest:

```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: HelloWorld Plug-in
Bundle-SymbolicName: example.HelloWorld
Bundle-Version: 1.0.0
Bundle-Activator: example.helloworld.Activator
Bundle-Vendor: DVS
Import-Package: org.osgi.framework;version="1.3.0"
```

* based on example from JavaWorld

Exporting and Importing Packages



TECHNISCHE
UNIVERSITÄT
DARMSTADT

```
package example.service;  
public interface HelloService {  
    public String sayHello();  
}
```

```
package example.service.impl;  
public class HelloServiceImpl implements HelloService {  
    public String sayHello() {  
        System.out.println("Inside HelloServiceImpl.sayHello()");  
        return "Say Hello";  
    }  
}
```

Add `Export-Package: example.service` to a bundle's manifest

Import in HelloWorld-Bundle: `Import-Package: example.service`

Defining Services

It is possible to register services:

```
context.registerService („hello“, helloService, null);
```

And get a reference somewhere else:

```
serviceRef = context.getServiceReference („hello“);  
HelloService helloService = (HelloService) context.getService (serviceRef);
```

Multiple implementations for the same name are possible.

**Is the OSGi framework a
component model or a service model?**

Announcements

Next paper + questions will be released in 3 weeks.

Use this time to **work on the project!**

Open office hours Tuesday, 16.12.2014

Questions



TECHNISCHE
UNIVERSITÄT
DARMSTADT

