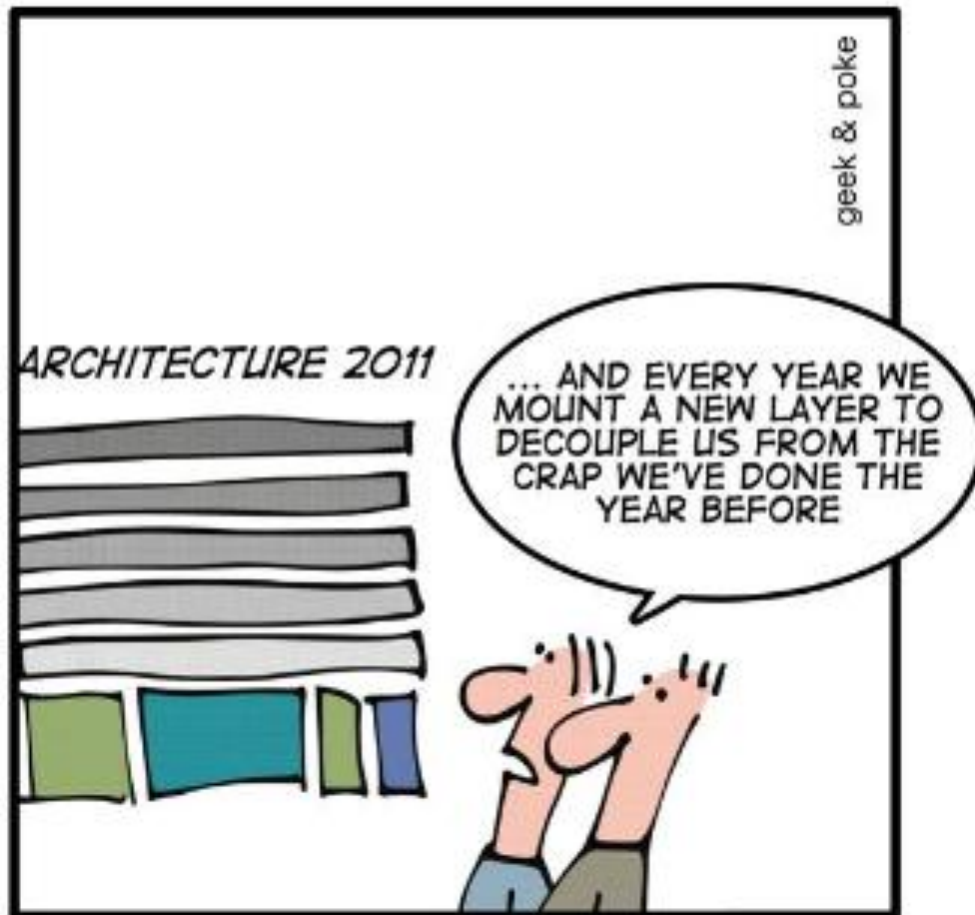


BEST PRACTICES IN APPLICATION ARCHITECTURE

TODAY: USE LAYERS TO DECOUPLE



OSGI & Spring

Enough rope to hang yourself

Alexander Koderman
SerNet GmbH

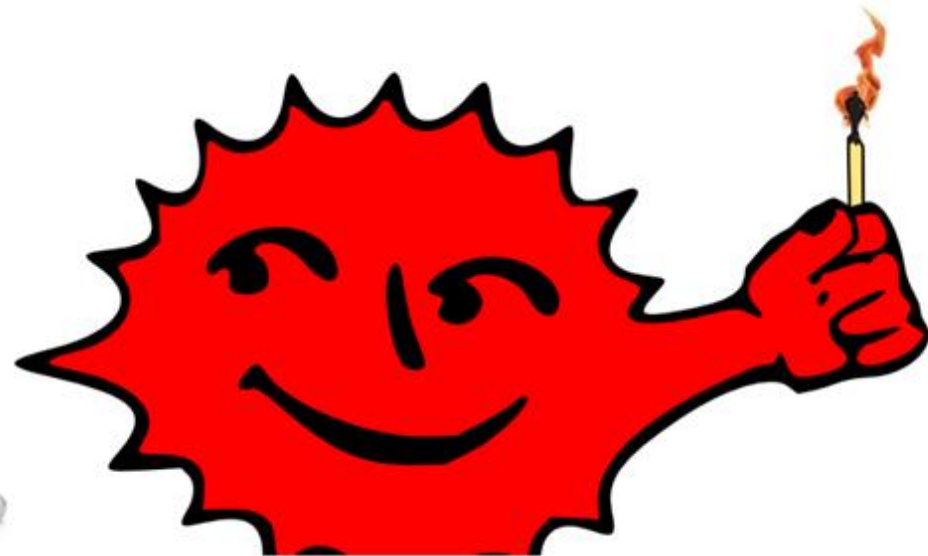
2011-04-07

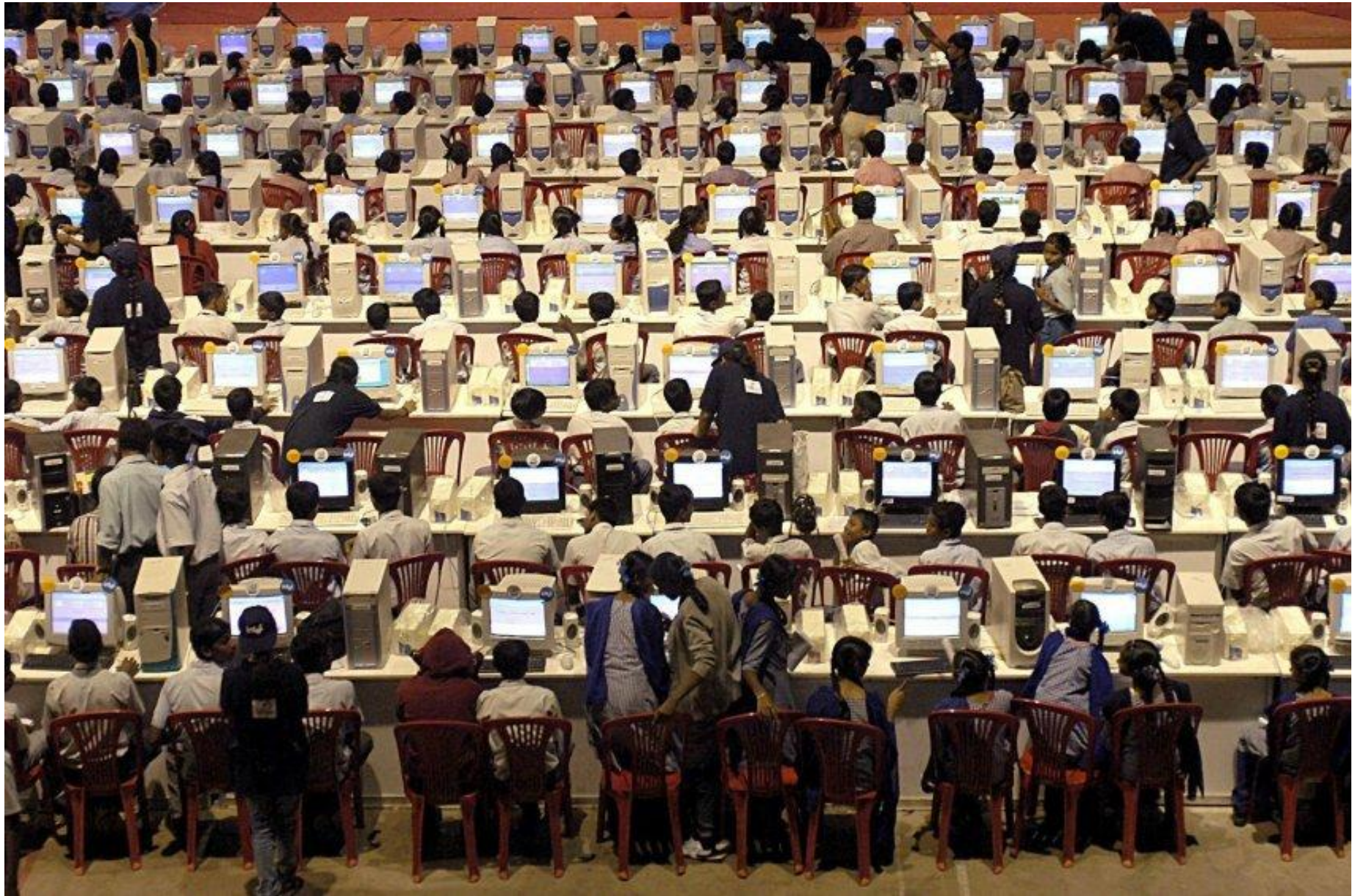
SerNet

- gegründet 1997
- Büros in Göttingen, Berlin, Nürnberg, Sunnyvale (Kalifornien)
- Informationssicherheit und Datenschutz
- spezialisiert auf Open Source Software (Samba)
- Zertifizierungen und Audits



Kernenergie.
So sicher wie diese Webseite.





Der Verband

Arbeitsgebiete & Themen

Zahlen & Fakten

Termine & Aktionen

Meldungen & Presse

Publikationen

Suchen



Startseite > Arbeitsgebiete & Themen > Informationsschutz und Risk Management > Weitere Informationen

VDA

Verband der
Automobilindustrie

Informationsschutz und Risk Management:

Weitere Informationen

Weitere Informationen

VDA ISA Tool verinice



[Link](#)

15.12.10

Information Security Assessment Das ISA-Tool verinice

Der Fragenkatalog zur Informationssicherheit mit integriertem Prozessreifegradmodell "Information Security Assessment" wurde in dem Open-Source-Tool ...

[Details](#)

Ihr Ansprechpartner



Wolfgang Bartsch

Leiter Fachbereich
Inhousekommunikation

+49 30 897842-180

E-Mail: bartsch@vda.de

Bitte senden Sie den ausgefüllten Fragebogen per Mail zurück an BenchmarkInfSchutz@vda.de

verinice.

verinice. [verinice.PRO](#) [verinice.XP](#) [verinice.FORUM](#)

Schrift: - +

News

Details

Service & Support

Screenshots

FAQ

Partner

Download



supported by

SerNet

licensed by

Bundesamt
für Sicherheit in der
Informationstechnik



verinice. Screenshots



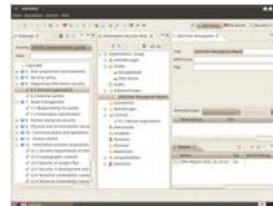
Import von CSV-Dateien



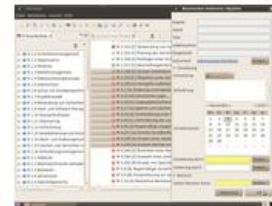
Verwaltung von Dokumenten in verinice



Ansicht in der Grundschutz-Perspektive



Ansicht in der Perspektive für ISO 27001



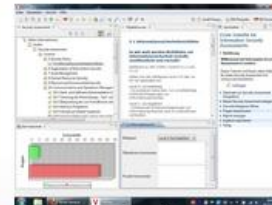
Masseneditor von Maßnahmen



Erstellen von Reports



Automatisierte individuelle Risikoanalysen



Information Security Assessment des VDA unter Windows 7



Report: Risiko Matrix (ISO 27005)

Newsletter

Um regelmäßig über neue Versionen und Änderungen informiert zu werden oder sonstige spannende Informationen zu verinice zu erhalten, melden Sie sich einfach bei unserem [Newsletter](#) an.

Software...

- wird entwickelt
- ...statt produziert
- Übertragung des Komponentengedankens auf Software 1968 durch Doug McIlroy

10

Summary--what's most important.

To put my strongest concerns in a nutshell:

1. We should have some ways of coupling programs like garden hose--screw in another segment when it becomes when it becomes necessary to massage data in another way. This is the way of IO also.
2. Our loader should be able to do link-loading and controlled establishment.
3. Our library filing scheme should allow for rather general indexing, responsibility, generations, data path switching.
4. It should be possible to get private system components (all routines are system components) for bugging around with.

M. D. McIlroy
Oct. 11, 1964

Anforderungen an eine SW-Komponente

- Kompartimentalisierung
- Kompositionsfähigkeit
- Repräsentation ausgewählter Dienste nach Außen
 - über Zugangspunkte
 - benötigte Dienste (Kontextbedingungen)
 - Ortstransparenz (Trennung von Schnittstelle und Implementation)
 - *nicht immer positiv*

Anforderungen an eine SW-Komponente

- Komponente am nützlichsten, wenn sie
 - viele Dienste anbietet
 - keinerlei Kontextbedingungen hat
 - überall einsetzbar ist
 - Balanceakt zwischen Unabhängigkeit und Gewicht
- Vertrag (in maschinenlesbarer Form)
 - funktionale Vereinbarungen
 - semantische Vereinbarungen
 - (Versionierung)

Anforderungen an eine SW-Komponente

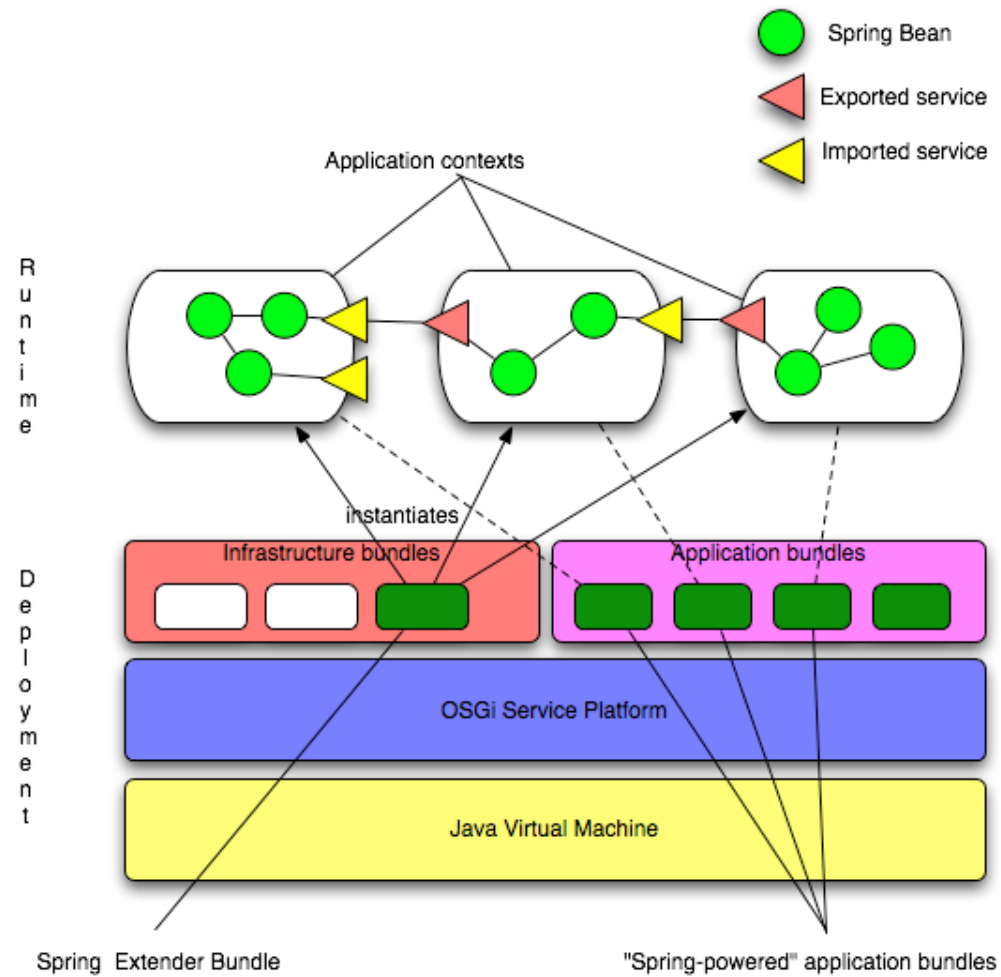
■ Mapping Decision

- Abhängigkeit des Clients von Implementierung
- Einflussnahme des Clients?
- „Coding between the lines“ (Kiczales)

Anforderungen an eine SW-Komponente

- objektorientierte Systeme:
 - feine Granularität
 - sehr schlank
 - explodierende Kontextabhängigkeiten
- daher:
 - Unterscheidung zwischen fein- mittel- und grobkörnigen Komponenten nötig
 - (Griffel, Szyperski...)
- Komponentenarchitektur nötig, die mehrere Granularitätsstufen abdeckt

OSGI und Spring



OSGI und Spring

- JAR-Files als Komponenten?
 - Illusion
 - „Deployment-time convenience“
- OSGI bundles & Spring beans

The screenshot shows the top portion of the IBM developerWorks website. At the top left is the IBM logo. To the right, there is a language selector set to "English" and a "Sign in" link. Below this is a dark orange navigation bar containing the "developerWorks" logo and several menu items: "Technical topics", "Evaluation software", "Community", and "Events". A search box is partially visible on the right side of this bar. Below the navigation bar, a breadcrumb trail reads "developerWorks > Open source > Technical library >". The main content area features a large heading: "Leave Eclipse plug-in headaches behind with OSGi", followed by a sub-heading: "Use the Open Services Gateway Initiative API to eliminate Eclipse plug-in dependencies".

Using 3rd party bundles

- Abhängigkeit von Benutzung der ClassLoader
- Hibernate sucht Model Klassen zur Laufzeit
- findet nix
 - Lösung 1: Implement your own SessionFactory
 - Lösung 2: Eclipse-RegisterBuddy

Good buddies:

- sernet.gs.service/META-INF/MANIFEST.MF:Eclipse-RegisterBuddy:
sernet.springclient,com.springsource.org.hibernate
- sernet.gs.ui.rcp.main/META-INF/MANIFEST.MF:Eclipse-RegisterBuddy:
sernet.verinice.oda.driver,com.springsource.org.hibernate
- sernet.verinice.report.service/META-INF/MANIFEST.MF:Eclipse-RegisterBuddy:
sernet.verinice.oda.driver
- sernet.verinice.samt.service/META-INF/MANIFEST.MF:Eclipse-RegisterBuddy:
com.springsource.org.hibernate
- target-platform/eclipse-3.5.2-rcp-sdk/plugins/inetlib/META-INF/MANIFEST.MF:Eclipse-RegisterBuddy:
sernet.verinice.encryption,com.springsource.javax.activation
- target-platform/eclipse-3.5.2-rcp-sdk/plugins/bouncycastle/META-INF/MANIFEST.MF:Eclipse-RegisterBuddy: gnumail



Hibernate Core / HHH-2318

Sometimes wrong classloader used for proxy interfaces

[Log In](#)

Details

Type:	Bug	Status:	Open
Priority:	Major	Resolution:	Unresolved
Affects Version/s:	3.2.1	Fix Version/s:	None
Component/s:	core		
Participants:	Jan Wiemer, Lars J. Nilsson and Max Rydahl Andersen		

Description:

▼ For some of our business classes we used a Mapping declaring the class to be lazy initialized and providing a proxy interface like e.g.:

```
<class name="TestClassImpl" proxy="TestClass" table="testTable" lazy="true"> ... </class>
```

Using classes mapped this way - e.g. as endpoint of a one to one relation - sporadically leads to exceptions like the following:

```
org.hibernate.PropertyAccessException: IllegalArgumentException occurred while calling setter of test.AbstractTestClass2Impl.testField
at org.hibernate.property.BasicPropertyAccessor$BasicSetter.set(BasicPropertyAccessor.java:104)
at org.hibernate.tuple.entity.AbstractEntityTuplizer.setPropertyValues(AbstractEntityTuplizer.java:337)
at org.hibernate.tuple.entity.PojoEntityTuplizer.setPropertyValues(PojoEntityTuplizer.java:200)
at org.hibernate.persister.entity.AbstractEntityPersister.setPropertyValues(AbstractEntityPersister.java:3514)
at org.hibernate.engine.TwoPhaseLoad.initializeEntity(TwoPhaseLoad.java:129)
at org.hibernate.loader.Loader.initializeEntitiesAndCollections(Loader.java:842)
at org.hibernate.loader.Loader.doQuery(Loader.java:717)
at org.hibernate.loader.Loader.doQueryAndInitializeNonLazyCollections(Loader.java:224)
at org.hibernate.loader.Loader.doList(Loader.java:2211)
at org.hibernate.loader.Loader.listIgnoreQueryCache(Loader.java:2095)
at org.hibernate.loader.Loader.list(Loader.java:2090)
at org.hibernate.loader.hql.QueryLoader.list(QueryLoader.java:388)
at org.hibernate.hql.ast.QueryTranslatorImpl.list(QueryTranslatorImpl.java:338)
```

Due to a compromise, all users are being forced to change their passwords as a precaution. You will need to use the "Can't access your account?" link next to the Log In button to request a reset email.



Hibernate Core / HHH-2317

CGLIB unable to lazy load classes in diff. classloader than hibernate jars

[Log In](#)

Details

Type:	Bug	Status:	Open
Priority:	Major	Resolution:	Unresolved
Affects Version/s:	3.2.1	Fix Version/s:	None
Component/s:	core		
Participants:	Jason Novotny, Martin Kneissl, Max Rydahl Andersen, Mike Meessen and Pietu Pohjalainen		

Description:

Please see <http://forum.hibernate.org/viewtopic.php?p=2334617#2334617> for more details and possible fix

Attachments

	hibernate-3.1.patch (1 kB) Pietu Pohjalainen	15/Jan/08 6:16 AM
	HibernateSpecifyClassLoaderToCGLIB.patch (0.5 kB) Mike Meessen	27/Apr/09 3:35 AM

Issue Links

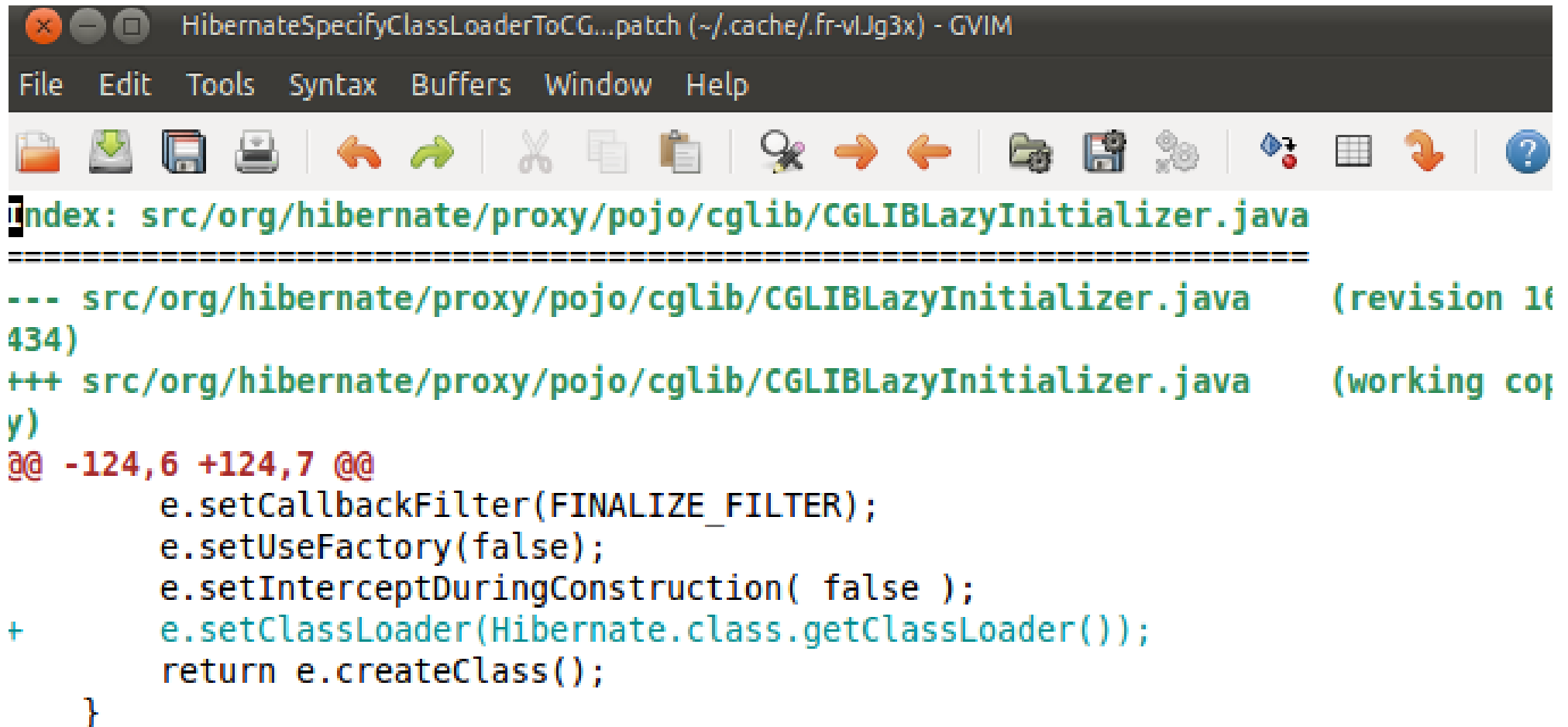
Duplicate

This issue *is duplicated by*:

HHH-2318 Sometimes wrong classloader used for proxy interfaces



Forcing a „Mapping Decision“



```
index: src/org/hibernate/proxy/pojo/cglib/CGLIBLazyInitializer.java
-----
--- src/org/hibernate/proxy/pojo/cglib/CGLIBLazyInitializer.java      (revision 16
434)
+++ src/org/hibernate/proxy/pojo/cglib/CGLIBLazyInitializer.java      (working copy)
@@ -124,6 +124,7 @@
     e.setCallbackFilter(FINALIZE_FILTER);
     e.setUseFactory(false);
     e.setInterceptDuringConstruction( false );
+    e.setClassLoader(Hibernate.class.getClassLoader());
     return e.createClass();
 }
```


Import-package vs. Require- Bundle



Import-package vs. Require- Bundle

The screenshot displays the Eclipse IDE's Dependencies view, which is divided into two main sections: Required Plug-ins and Imported Packages.

Required Plug-ins: This section is titled "Required Plug-ins" and includes a sub-header "Specify the list of plug-ins required for the operation of this plug-in." It contains a list of 15 plug-ins, each with a small icon and its version number. The list is as follows:

- org.eclipse.equinox.ds (1.1.1)
- org.eclipse.equinox.app (1.2.1)
- org.eclipse.equinox.common (3.5.1)
- org.eclipse.equinox.preferences (3.2.301)
- org.eclipse.core.jobs (3.4.100)
- org.eclipse.ui.workbench (3.5.2)
- org.eclipse.jface (3.5.2)
- org.eclipse.swt (3.5.2)
- org.eclipse.ui.cheatsheets (3.3.200)
- org.eclipse.core.resources (3.5.2)
- org.eclipse.core.net (1.2.1)
- org.eclipse.ui.ide (3.5.2)
- org.eclipse.ui.intro (3.3.2)
- org.eclipse.core.expressions (3.4.101)
- org.springframework.beans (2.5.6)
- org.springframework.context (2.5.6)

At the bottom right of this list, it says "Total: 45". To the right of the list are several control buttons: "Add...", "Remove", "Up", "Down", and "Properties...".

Imported Packages: This section is titled "Imported Packages" and includes a sub-header "Specify packages on which this plug-in depends without explicitly identifying their originating plug-in." It contains a list of 15 packages, each with a small icon and its version number. The list is as follows:

- javax.xml.rpc.encoding
- org.apache.axis.encoding.ser
- org.eclipse.core.runtime (3.4.0)
- org.eclipse.osgi.service.datalocation (1.2.0)
- org.eclipse.osgi.util (1.1.0)
- org.jaxen
- org.jdom
- org.jdom.input
- org.jdom.output
- org.jdom.xpath
- org.jfree.chart
- org.jfree.chart.axis
- org.jfree.chart.labels
- org.jfree.chart.plot
- org.jfree.chart.renderer.category
- org.jfree.chart.renderer.xy

At the bottom right of this list, it says "Total: 25". To the right of the list are several control buttons: "Add...", "Remove", and "Properties...".

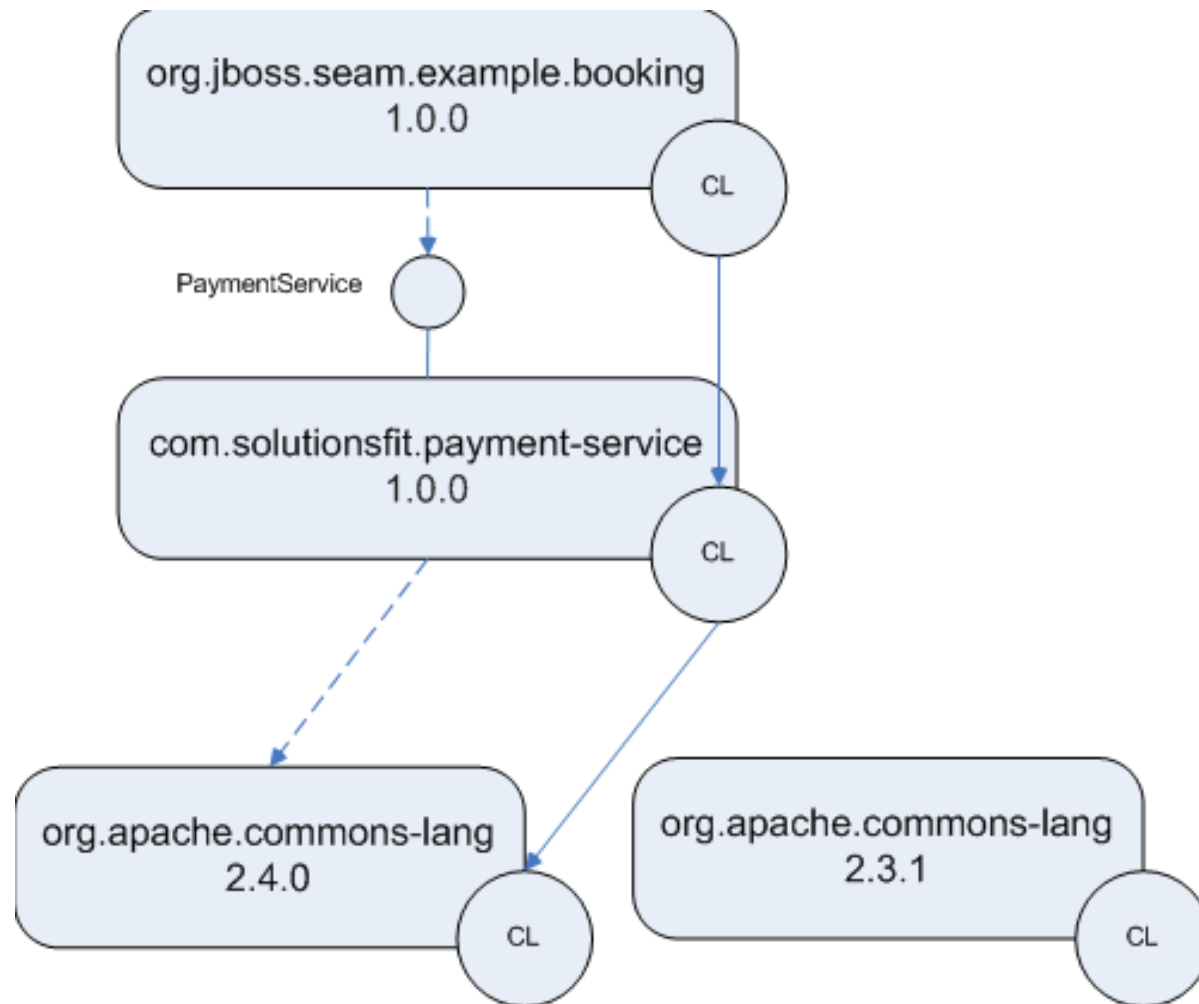
Import-package vs. Require- Bundle

- Import-Package: lose Kopplung
- nicht immer eigene Entscheidung (3rd party bundles)
- Import-package: Abhängigkeit weniger offensichtlich
- Granularität?

Import-package vs. Require- Bundle

- Import-Package: lose Kopplung
- nicht immer eigene Entscheidung (3rd party bundles)
- Import-package: Abhängigkeit weniger offensichtlich
- Granularität?
- Superklassen nicht vergessen:
 - adding Hibernate.STRING
 - STRING ist eine Subklasse von org.hibernate.type.Type
 - ClassNotFoundException zur Laufzeit

Import-package vs. Require- Bundle



Import-package vs. Require- Bundle

The Google logo is displayed in its characteristic multi-colored font (blue, red, yellow, blue, green, red).

import package uses conflict

About 5,670,000 results (0.26 seconds)

Package Uses Conflict

- *„If the same fully qualified class name is defined by two distinct class loaders, then this produces two incompatible runtime types.“ (springsource.org)*
- Das bedeutet:
 - Ein Bundle importiert eine Klasse immer nur aus **einem** Bundle.
 - Trotzdem kann eine Klasse implizit über ein anderes Bundle importiert werden:
 - *„Type A refers to type B which refers to another type C, a user of A can obtain a reference to C via B.“*
 - Steuern über die „uses“ Direktive

This fails always:



This fails sometimes:





Minimize Dependencies

Best Practice

- Use **Import-Package** instead of **Require-Bundle**
 - Require-Bundle can have only one provider—the named bundle
 - Import-Package can have many providers
 - Allows for more choices during resolving
 - Has a lower fan out, which gain adds up quickly
- Use version ranges
 - Using precise version numbers gives the dependency resolver less choice
- Design your bundles
 - Don't put unrelated things in the same bundle
 - Low coupling, high cohesion

Package Uses Conflict

- *„If the same fully qualified class name is defined by two distinct class loaders, then this produces two incompatible runtime types.“ (springsource.org)*
- *“Uses constraints are common in enterprise libraries, and manually diagnosing a failure can be a real nightmare. In particular, determining the possible conflicts can be extremely time-consuming when you have an exported package with 10 or more packages listed in its uses clause.“*

Services...

- *„Dont assume that if you are having classloader problems or resolution problems that you can reload bundles. I have found the reloading bundles only works reliably when there are no classloader or resolution problems. When there are, its anyones guess what state the OSGi container is left in. I have been sutting down, and completely deleiting all state as the only reliable way of debugging dependencies“ (blog.tfd.co.uk)*
- other Gotchas...

Spring (not Spring DM...)

- Dependency Injection
- mehr Flexibilität
- weniger Boilerplate Code
- *„One drawback is that excessive or inappropriate use of dependency injection can make applications more complicated, harder to understand, and more difficult to modify.“* (Wikipedia)
- konkret:
 - Interne Abhängigkeiten müssen bekannt sein
 - Dokumentation unzureichend / veraltet

Beispiel: HTTP Digest Authentication

- *FilterChainProxy*
 - *HttpSessionContextIntegrationFilter*
 - *DigestProcessingFilter*
 - ExceptionTranslationFilter
 - FilterSecurityInterceptor

Beispiel: HTTP Digest Authentication

- *FilterChainProxy*
 - *HttpSessionContextIntegrationFilter*
 - *DigestProcessingFilter*
 - *DigestProcessingFilterEntryPoint*
 - *ExceptionTranslationFilter*
 - *DigestProcessingFilterEntryPoint*
 - *FilterSecurityInterceptor*
 - *security:authenticationManager*
 - *accessDecisionManager (AffirmativeBased)*
 - *RoleVoter*

Beispiel: HTTP Digest Authentication

- *FilterChainProxy*
 - *HttpSessionContextIntegrationFilter*
 - *DigestProcessingFilter*
 - *DigestProcessingFilterEntryPoint*
 - *DbUserDetailsService* <----- You are here
 - *loadUserConfigurationCommand*
 - "commandService"
 - *ExceptionTranslationFilter*
 - *DigestProcessingFilterEntryPoint*
 - *FilterSecurityInterceptor*
 - *security:authenticationManager*
 - *accessDecisionManager (AffirmativeBased)*
 - *RoleVoter*

Danke für Ihre Aufmerksamkeit!

verinice.[®]

<http://www.verinice.org>

<http://v.de>

SerNet GmbH

Bahnhofsallee 1b

37081 Göttingen

Tel: +49 -551-370000-0

Fax: +49 -551-370000-9

<http://www.SerNet.DE>

Schützenstr. 18

10117 Berlin

+49 -30 -5 779 779-0

+49 -30 -5 779 779-9