

5.– 8. September 2011
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Wolkige Angelegenheit

Rapid Cloud Development mit Spring Roo

Kai Wähler

MaibornWolff et al GmbH



Nur wenige Folien ...

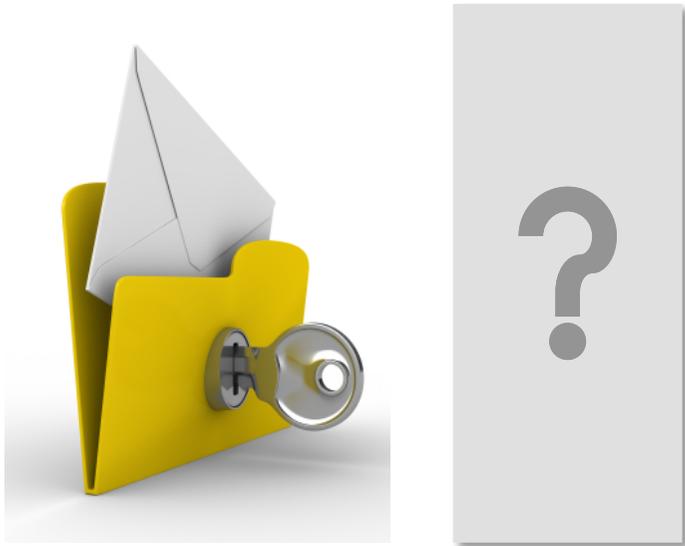
... stattdessen viel „Live Hacking“ !!!





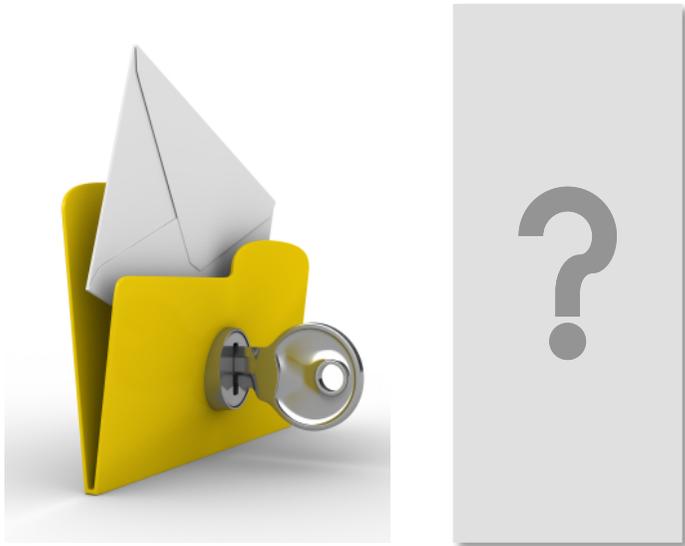


Die Konzepte von Spring Roo und PaaS verstehen!



Die Konzepte von Spring Roo und PaaS verstehen!

Die Kombination von Spring Roo und PaaS kennenlernen!



Die Konzepte von Spring Roo und PaaS verstehen!

Die Kombination von Spring Roo und PaaS kennenlernen!

Sinnvolle Einsatzmöglichkeiten verstehen!

1) Spring Roo

2) Platform as a Service (Paas)

3) Google App Engine (GAE)

4) Live Hacking: GAE mit Spring Roo

5) VMware Cloud Foundry

6) Live Hacking: Cloud Foundry mit Spring Roo

7) Kritik und Einsatzempfehlungen für Spring Roo

1) Spring Roo

2) Platform as a Service (Paas)

3) Google App Engine (GAE)

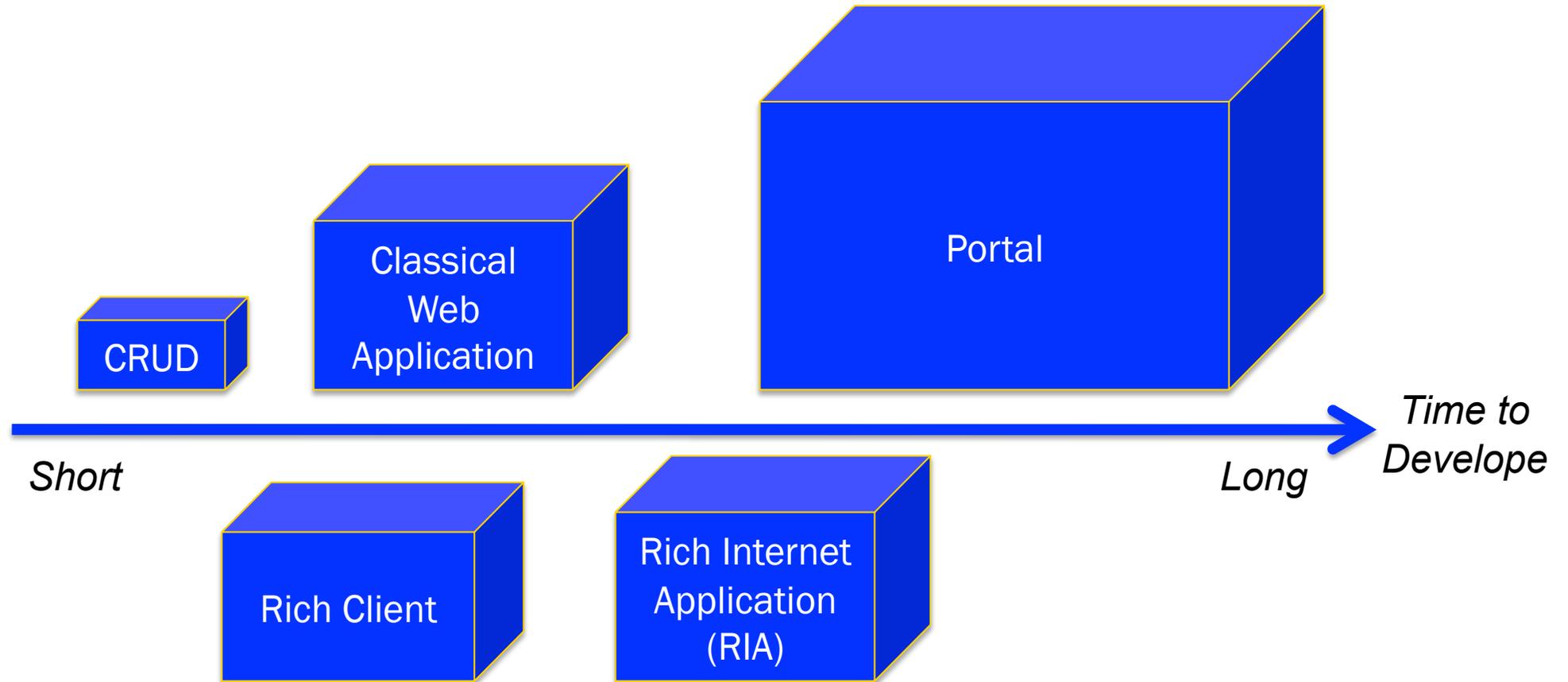
4) Live Hacking: GAE mit Spring Roo

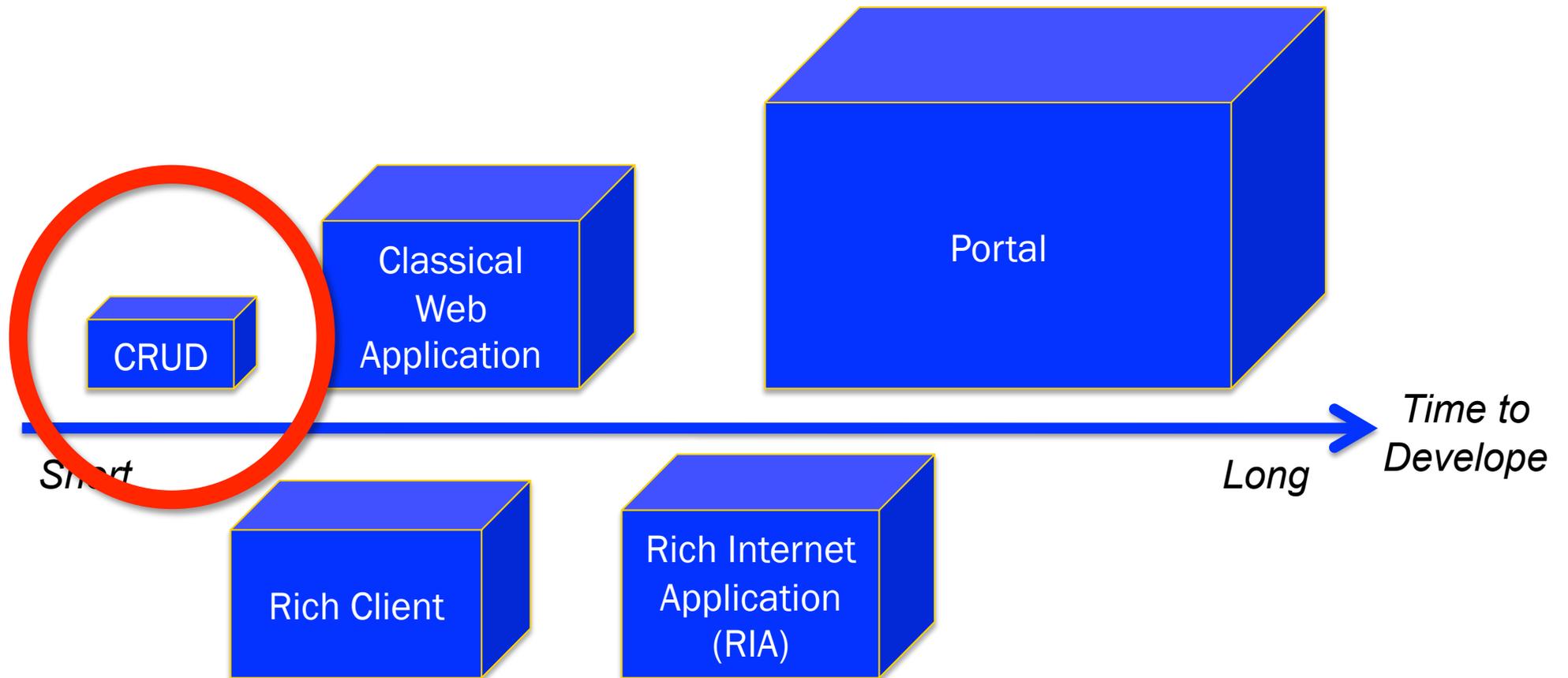
5) VMware Cloud Foundry

6) Live Hacking: Cloud Foundry mit Spring Roo

7) Kritik und Einsatzempfehlungen für Spring Roo

Rapid Application Development









(Roma Framework)



Gutes *CRUD* Framework



(Roma Framework)



Gutes CRUD Framework



(Roma Framework)



Kein Java



Gutes CRUD Framework



(Roma Framework)

Kein großer Unterstützer

Keine große Community



Kein Java



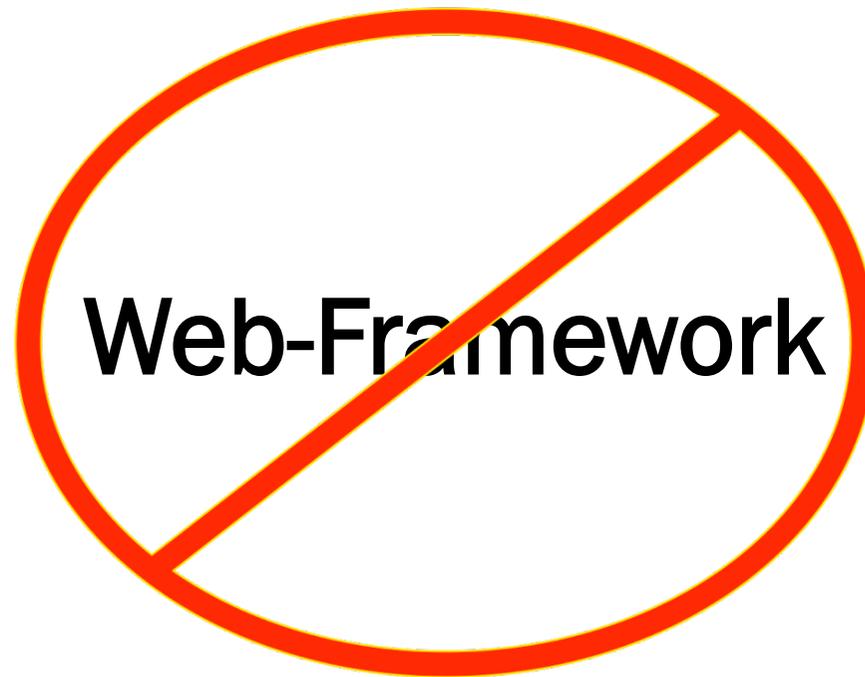
GRAILS



... und der Gewinner* ist:



* aus der Sicht eines Java-Entwicklers!



Full-Stack-Framework





**Configuration
Marathon**

Roo-Shell



Java

Maven

Spring

JPA

AspectJ

JUnit

JMS

Vaadin

Solr

GWT

Selenium

**Anzahl
kontinuierlich
steigend**

Weitere ...





AspectJ

Roo-Shell

Spring



AspectJ

Roo-Shell

Spring



AspectJ

Roo-Shell

Spring



AspectJ

Roo-Shell

Spring

1) Spring Roo

2) Platform as a Service (Paas)

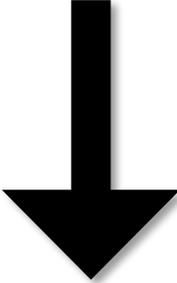
3) Google App Engine (GAE)

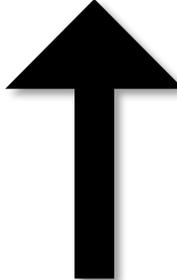
4) Live Hacking: GAE mit Spring Roo

5) VMware Cloud Foundry

6) Live Hacking: Cloud Foundry mit Spring Roo

7) Kritik und Einsatzempfehlungen für Spring Roo

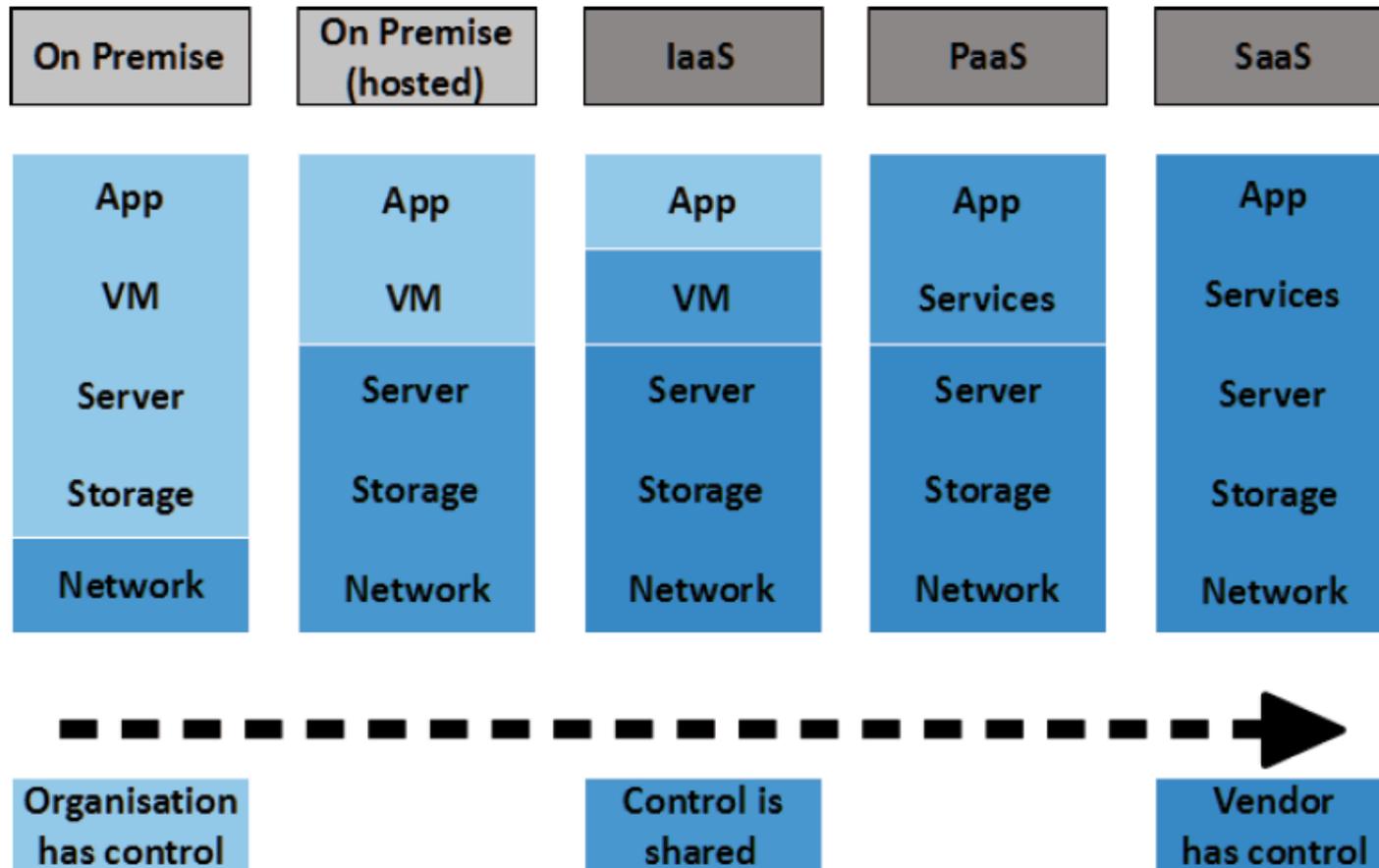
Kosten 

Flexibilität 

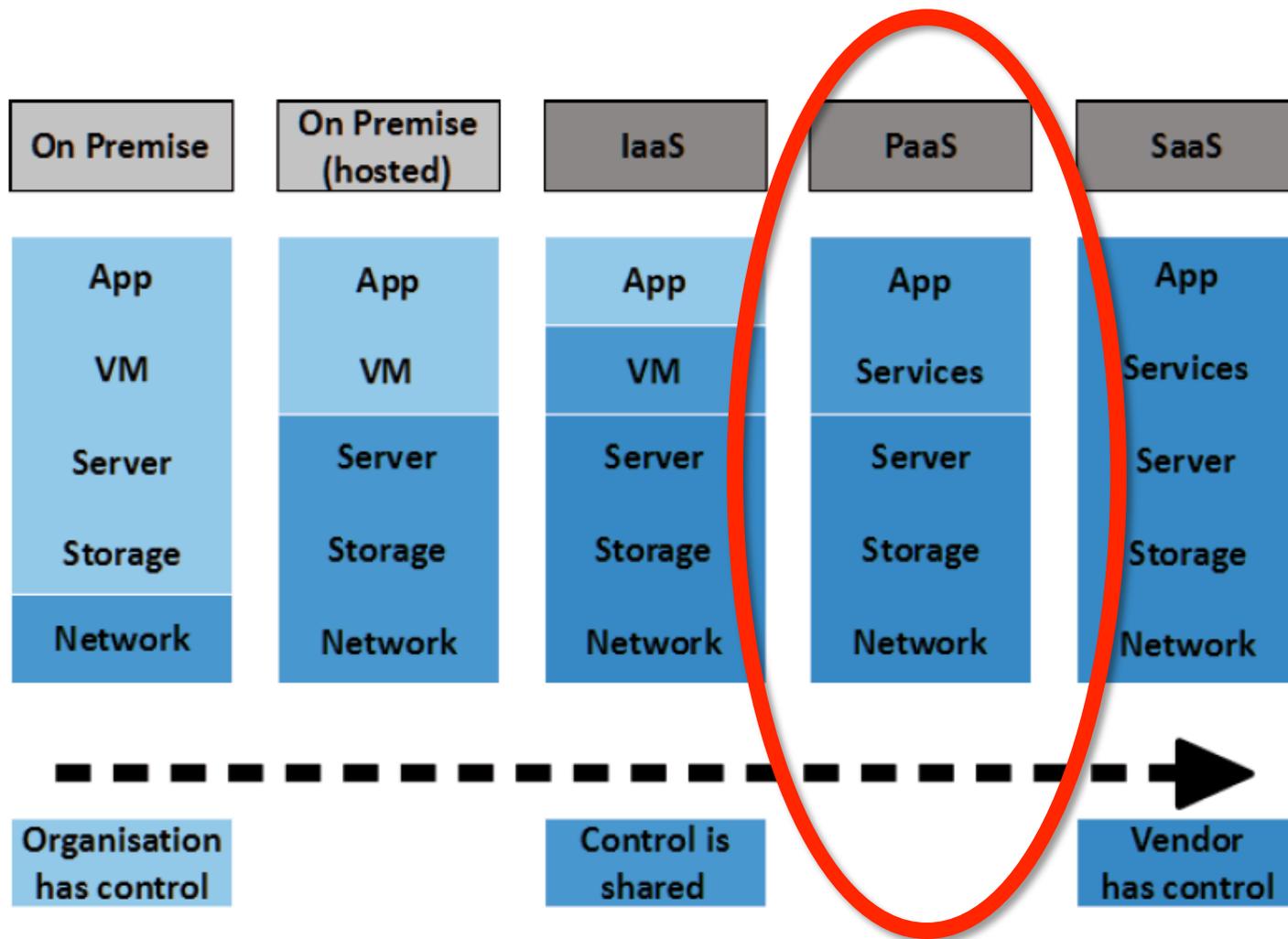
Private Cloud

Hybrid Cloud

Public Cloud



Cloud-Security and Privacy, O'Reilly, 2009



Cloud-Security and Privacy, O'Reilly, 2009

„Platform as a service (PaaS) is the delivery of a computing platform and solution stack as a service. PaaS offerings facilitate deployment of applications without the cost and complexity of buying and managing the underlying hardware and software and provisioning hosting capabilities.

[PaaS provides] all of the facilities required to support the complete life cycle of building and delivering web applications and services entirely available from the Internet.“

Wikipedia

Einfaches Deployment

vs.

Einschränkungen bei der Entwicklung



Anzahl kontinuierlich steigend ...



Anzahl kontinuierlich steigend ...

1) Spring Roo

2) Platform as a Service (Paas)

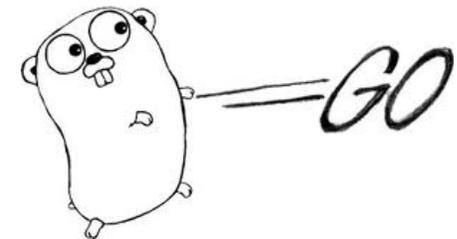
3) Google App Engine (GAE)

4) Live Hacking: GAE mit Spring Roo

5) VMware Cloud Foundry

6) Live Hacking: Cloud Foundry mit Spring Roo

7) Kritik und Einsatzempfehlungen für Spring Roo



Private Cloud

Hybrid Cloud



Public Cloud



Push Queue

Pull Queue

URL Fetch

Accounts

Mail

Memcache

XMPP

Images

Datastore

Blobstore

JRE Class White List

Workarounds für Frameworks

No „naked“ Domains

Eingeschränkte Portabilität

Keine SQL-Datenbank



**JPA / JDO /
low level API**

1) Spring Roo

2) Platform as a Service (Paas)

3) Google App Engine (GAE)

4) Live Hacking: GAE mit Spring Roo

5) VMware Cloud Foundry

6) Live Hacking: Cloud Foundry mit Spring Roo

7) Kritik und Einsatzempfehlungen für Spring Roo

1) Spring Roo

2) Platform as a Service (Paas)

3) Google App Engine (GAE)

4) Live Hacking: GAE mit Spring Roo

5) VMware Cloud Foundry

6) Live Hacking: Cloud Foundry mit Spring Roo

7) Kritik und Einsatzempfehlungen für Spring Roo



Private Cloud

Hybrid Cloud

Public Cloud



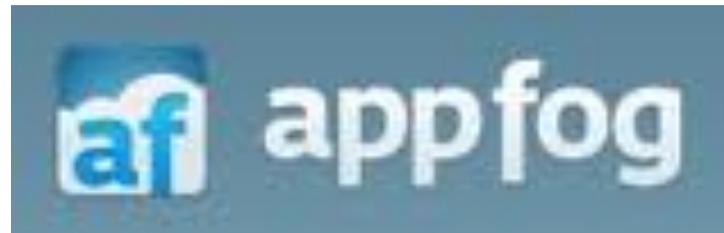


stackato™



PostgreSQL





Nur die unterstützen Services nutzbar

**Spring-Anwendung
(ansonsten deutlich mehr Aufwand)**

noch BETA

1) Spring Roo

2) Platform as a Service (Paas)

3) Google App Engine (GAE)

4) Live Hacking: GAE mit Spring Roo

5) VMware Cloud Foundry

6) Live Hacking: Cloud Foundry mit Spring Roo

7) Kritik und Einsatzempfehlungen für Spring Roo

1) Spring Roo

2) Platform as a Service (Paas)

3) Google App Engine (GAE)

4) Live Hacking: GAE mit Spring Roo

5) VMware Cloud Foundry

6) Live Hacking: Cloud Foundry mit Spring Roo

7) Kritik und Einsatzempfehlungen für Spring Roo







```
[INFO]           Compiling permutation 11...
[INFO]     Compile of permutations succeeded
[INFO] Linking into /Users/User/Documents/workspace-sts-2.7.1.RELEASE/SimpleCloudFoundry/target/SimpleCloudFoundry-0.1.0.BUILD-SNAPSHOT/applicationSc
[INFO]   Link succeeded
[INFO]     Compilation succeeded -- 141.674s
[INFO]
[INFO] --- maven-war-plugin:2.1.1:war (default-war) @ SimpleCloudFoundry ---
[INFO] Packaging webapp
[INFO] Assembling webapp [SimpleCloudFoundry] in [/Users/User/Documents/workspace-sts-2.7.1.RELEASE/SimpleCloudFoundry/target/SimpleCloudFoundry-0.1.0
[INFO] Processing war project
[INFO] Copying webapp resources [/Users/User/Documents/workspace-sts-2.7.1.RELEASE/SimpleCloudFoundry/src/main/webapp]
[INFO] Webapp assembled in [2469 msecs]
[INFO] Building war: /Users/User/Documents/workspace-sts-2.7.1.RELEASE/SimpleCloudFoundry/target/SimpleCloudFoundry-0.1.0.BUILD-SNAPSHOT.war
[INFO] WEB-INF/web.xml already added, skipping
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----+-----
[INFO] Total time: 2:54.668s
[INFO] Finished at: Wed Jul 27 13:02:04 CEST 2011
[INFO] Final Memory: 8M/81M
[INFO] -----
Operation could not be completed: 400 Bad Request
```

```
----- Applications -----
```

Name	Status	Instances	Services	URLs
kaiwaehner1test	STARTED	1		kaiwaehner1test.cloudfoundry.com

```
~.entities.Project root> cloud foundry stop app --appName kaiwaehnertest1  
NullPointerException at org.springframework.roo.addon.cloud.foundry.CloudFoundryOperationsImpl  
$12.execute(CloudFoundryOperationsImpl.java:279)  
~.entities.Project root>
```





+



Production-ready

Sehr ausführliche Dokumentation

Stetige Weiterentwicklung
(SQL-Support „bald“ ?!)

Schnelle Realisierung von
CRUD-Anwendungen für die Cloud

Kostenlos (für kleine Anwendungen)

JPA (viele Einschränkungen) statt JDO

Relationen (noch ?!) per Hand umsetzen

Keine GAE-spezifischen Roo-Kommandos

Kein „Write once run everyhwere“ ☹



+



**Web-Anwendungen wie bisher entwickeln
und dann einfach (irgendwo) deployen**

MySQL => Relationale Datenbank

**Open Source / Community +
Kommerzielle Anbieter**

**Gute Integration von Cloud Foundry und
Roo in STS**

„Abhängigkeit“ von bestimmten Frameworks und Services

Kaum Dokumentation

Roo-Kommandos kein wirklicher Vorteil

Noch nicht produktiv einsetzbar / Kosten?

**Zugriff auf Cloud-DB nur über eigene Anwendung
(kein DB-Client oder Admin-GUI)**



=

Rapid Cloud Development
???

Ja, wenn ...

Ja, wenn ...

... das Ziel eine CRUD-Anwendung ist,

Ja, wenn ...

... das Ziel eine CRUD-Anwendung ist,

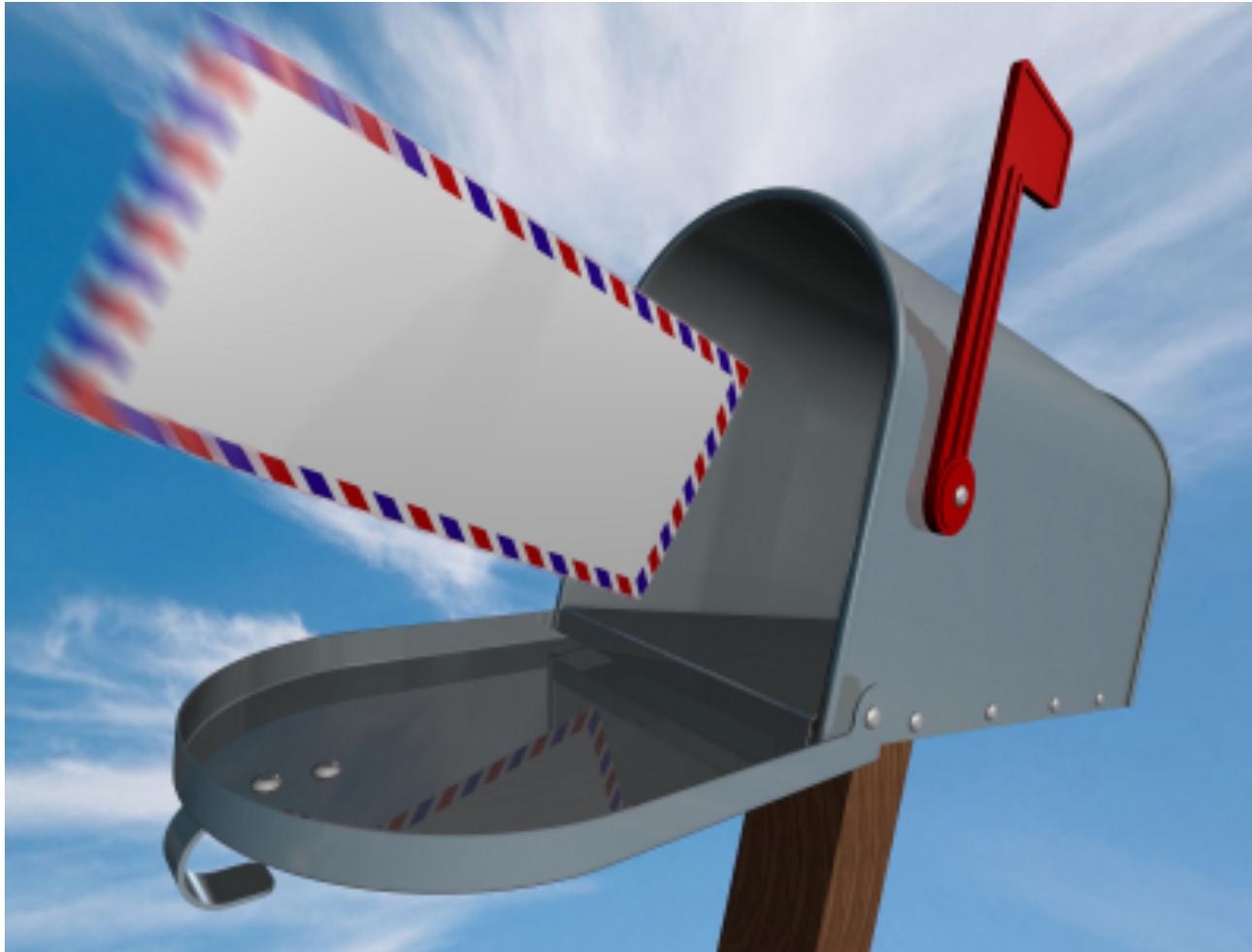
... und die Einschränkungen akzeptabel sind,

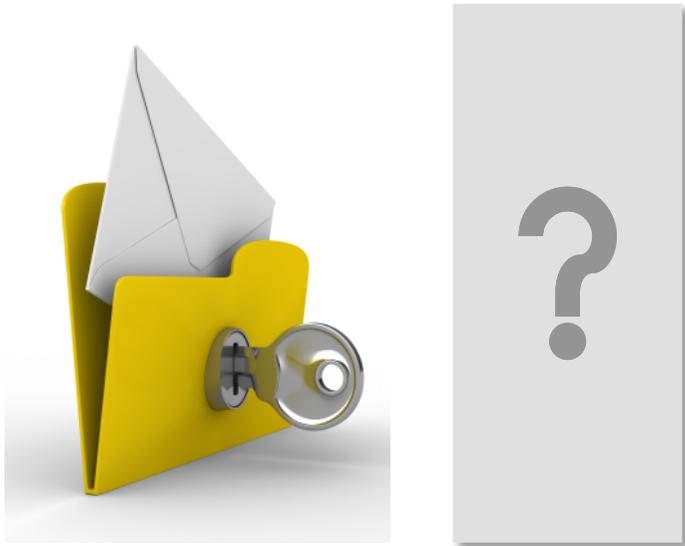
Ja, wenn ...

... das Ziel eine CRUD-Anwendung ist,

... und die Einschränkungen akzeptabel sind,

... **oder** man in PaaS-Entwicklung
einsteigen möchte.

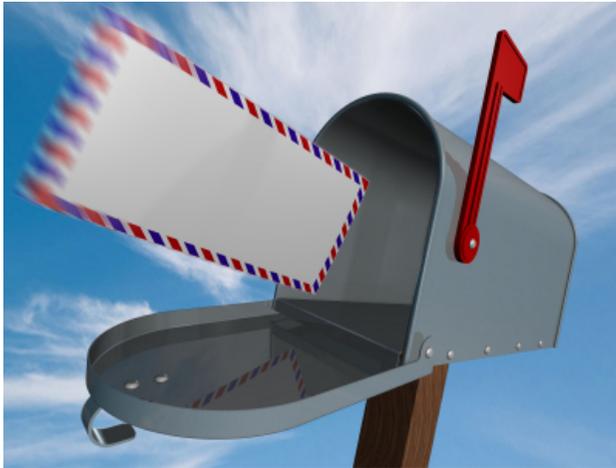


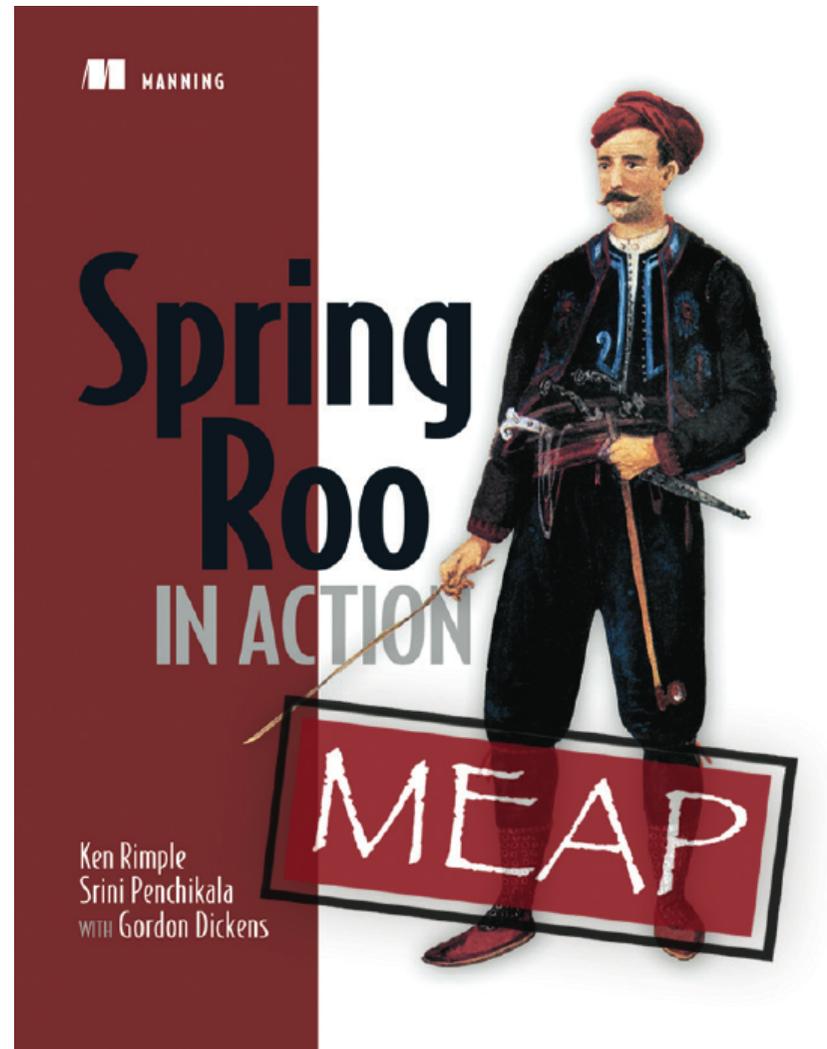


Die Konzepte von Spring Roo und PaaS verstehen!

Die Kombination von Spring Roo und PaaS kennenlernen!

Sinnvolle Einsatzmöglichkeiten verstehen!







Dashboards ▾ Projects ▾ Issues ▾

Spring Roo

- Summary
- Issues
- Road Map
- Change Log
- Popular Issues**
- Labels
- Versions
- Components
- Builds
- Source
- Reviews

Popular Issues

Shows the unresolved issues sorted by number of votes

111		ROO-120	Support for flat multi module project
85		ROO-301	Choose between data access patterns
81		ROO-91	Support Ant+Ivy as an alternative user project build system to Maven
81		ROO-516	JSF 2.0 Addon
75		ROO-163	Allow specification of a parent pom.
58		ROO-532	Spring Security addon
52		ROO-241	Allow user to specify "order by" on dynamic finders
49		ROO-442	Provide multipart file upload support
48		ROO-426	Spring Roo Addon for Spring Web Services
38		ROO-8	Review templating approach to be more flexible and allow better custom branding of generated applications
38		ROO-340	Introduce Service layer and move all finders on it

5.– 8. September 2011
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Vielen Dank für Ihre Aufmerksamkeit!

Kai Wähler

MaibornWolff et al GmbH

Email: kai.waehner@mwea.de

Twitter: [@KaiWaehner](https://twitter.com/KaiWaehner)