

Grails

Weiterführende Themen zu Internet- und WWW-Technologien

Matthias Springer

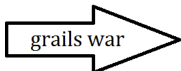
06. Juni 2011

- 1 Was ist Grails?
- 2 Übersicht über Grails
- 3 MVC-Konzept
- 4 Groovy Beispiele
- 5 Tutorial: Kleiner Notizblock

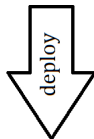
Grails Web Application Framework



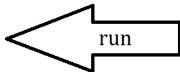
MyApplication
Grails Projekt



MyApplication-0.1.war
Web Archive (.war)



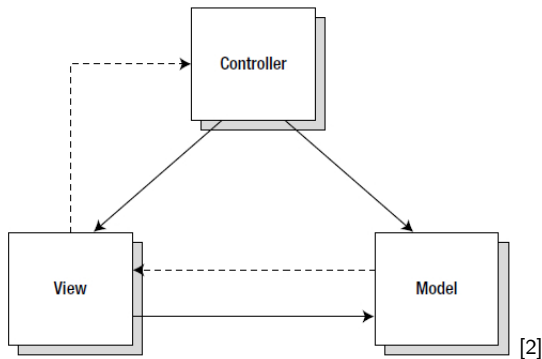
<http://localhost:8080/MyApplication-0.1/>
Browser



Apache Tomcat

- Groovy Programming Language
- Hibernate: object-relational mapping
- Hyperthreaded Structured Query Language Database (HSQLDB)
- Spring framework
- SiteMesh (*web-page layout and decoration framework*)
- Apache Tomcat / Jetty

MVC - Model, View, Controller



- **Model:** Datenstruktur + Geschäftslogik
- **View:** Präsentation, Darstellung der Daten
- **Controller:** Auswertung von Benutzeraktionen

- grails-app
 - conf: url mappings, bootstrapper, config
 - controllers: controller
 - domain: domain classes
 - i18n: Übersetzung
 - services: Grails services
 - views: GSPs (Groovy Server Pages), JSPs (Java Server Pages)
- lib: JAR-Bibliotheken
- src: sonstige Groovy- und Java-Quelldateien
- target: Ausgabeverzeichnis
- web-app: statische Ressourcen, z.B. Bilder, CSS, JS, ...

- `String textStatic = "Hallo Welt"`
- `def textDynamic = "Hallo Welt"`
- `assert textDynamic.length() == textStatic.length()`
- In Java nicht möglich

- Lambda-Ausdrücke
- `def closure = {x -> println 5 + x}; closure(5)`
- `def closure = {println 5 + it}; closure(5)`
- `def closure = {return it + 2}; println closure(5)`
- `def closure = {it + 2}; println closure(5)`
- In Java 6 nicht möglich

- `def list = ["Hello", 2, "World"]; list += [4, 5]`
- `[1, 2, 3].collect {it * it}`
- `assert ["Mo":"Montag", "Di":"Dienstag"].Mo == ["Mo":"Montag", "Di":"Dienstag"]["Mo"]`
- `(1..10).each {// for int it = 1 to 10}`
- `def tag = "Montag"; println "Heute ist ${tag}"`
- `(new GroovyShell()).evaluate("println 'Hello' + 'World'")`
- Reguläre Ausdrücke, Stringmanipulationen

Kleiner Notizblock

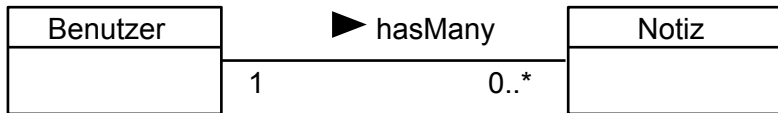


```
1 package notizblock
2
3 class Benutzer {
4     static hasMany = [notizen : Notiz]
5
6     String vorname
7     String anschrift
8     int alter
9 }
```

Benutzer.groovy

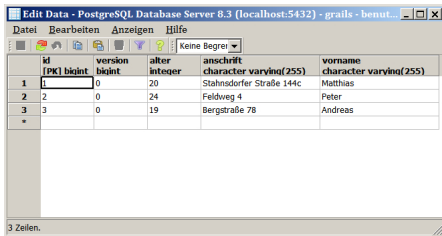
```
1 package notizblock
2
3 class Notiz {
4     static belongsTo = [ersteller : Benutzer]
5
6     String text
7     Date datum
8 }
```

Notiz.groovy



- `static belongsTo = [ersteller:Benutzer]`
Besitzer definieren
- `Benutzer ersteller`
- `static hasMany = [notizen:Notiz]`
*-to-many-Assoziation
- `java.util.Set<Notiz> notizen`
bei Delete keine Kaskadierung
- Referentielle Integrität immer sichergestellt

Grails Object Relational Mapping (GORM)



The screenshot shows a PostgreSQL database client window titled "Edit Data - PostgreSQL Database Server 8.3 (localhost:5432) - grails - benut...". The window displays a table with the following columns: id (PK), version, alter, anschrift, and vorname. The data is as follows:

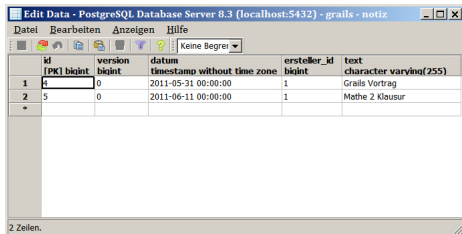
	id [PK] bigint	version bigint	alter integer	anschrift character varying(255)	vorname character varying(255)
1	1	0	20	Stahnsdorfer Straße 144c	Matthias
2	2	0	24	Feldweg 4	Peter
3	3	0	19	Bergstraße 78	Andreas
*					

At the bottom of the window, it says "3 Zeilen."

```
1 CREATE TABLE benutzer
2 (
3   id bigint NOT NULL,
4   "version" bigint NOT NULL,
5   "alter" integer NOT NULL,
6   anschrift character varying(255) NOT NULL,
7   vorname character varying(255) NOT NULL,
8   CONSTRAINT benutzer_pkey PRIMARY KEY (id)
9 )
```

benutzer.sql

Grails Object Relational Mapping (GORM)



	id [PK] bigint	version bigint	datum timestamp without time zone	ersteller_id bigint	text character varying(255)
1	1	0	2011-05-31 00:00:00	1	Grails Vortrag
2	5	0	2011-06-11 00:00:00	1	Mothe 2 Klausur
*					

2 Zeilen.

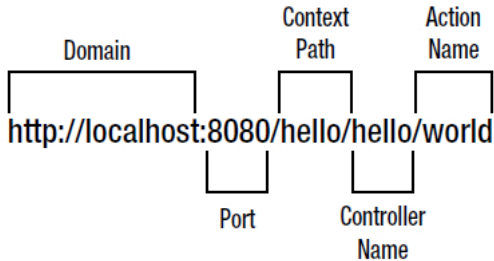
```
1 CREATE TABLE notiz
2 (
3   id bigint NOT NULL,
4   "version" bigint NOT NULL,
5   datum timestamp without time zone NOT NULL,
6   ersteller_id bigint NOT NULL,
7   "text" character varying(255) NOT NULL,
8   CONSTRAINT notiz_pkey PRIMARY KEY (id),
9   CONSTRAINT fk6424f44f5ad30d8 FOREIGN KEY (
10      ersteller_id)
11      REFERENCES benutzer (id) MATCH SIMPLE
```

notiz.sql

Controller

```
1 class HelloController {
2   def sayHello = {
3     render "Hello ${params.name} it's " + new java.util
4       .Date()
5   }
}
```

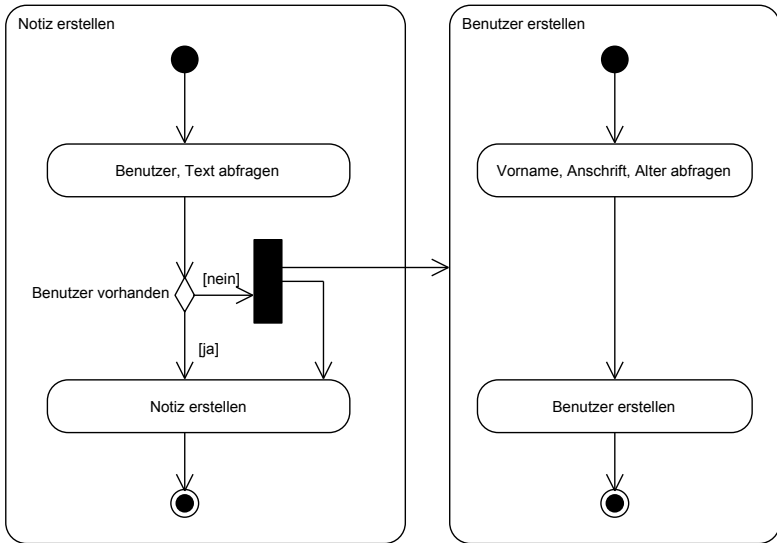
HelloController.groovy



[2]

- Parameterübergabe (für Debug-Zwecke)
 - HTTP POST
 - `http://localhost:8080/hello/hello/world?p1=1337`
 - `http://localhost:8080/hello/hello/world/512`
- Zugriff auf Parameter: `params`
- Zugriff Java Servlet-Schnittstellen: `request`, `response`, `servletContext`, `session` ^[5]

Controller: Ablaufdiagramm



- [1] Abdul-Jawad, B.: Groovy Grails Recipes. Apress, 2009.
- [2] Rocher, G. K.: The Definitive Guide to Grails. Apress, 2006.
- [3] Grails Quick Reference
<http://grails.org/doc/latest/ref/Constraints/Usage.html>
- [4] Grails Quick Reference
<http://grails.org/doc/latest/ref/Domain%20Classes/hasMany.html>
- [5] <http://download.oracle.com/javase/6/api/javax/servlet/http/HttpServletRequest.html>

- `homeEmail(email:true)`
- `name(inList:('Joe', 'Fred', 'Bob'))`
- `login(matches: '<<regex>>')`
- `age(max:100)`
- `age(nullable:true)`
- `age(range:18..100)`
- `login(unique:true)`
- `login(validator: {it.startsWith('user')})`