

12.–15.09.2010
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

K30

Kaffeeklatsch

Das Wichtigste in aller Kürze

Matthias Göttler

Hibernate Envers

„Easy Entity Auditing“

MATHEMA Software GmbH

Henkestraße 91

91052 Erlangen

info@mathema.de

www.mathema.de

...in 5 Minuten...?

...Hibernate...?

Keine Angst, es wird nicht wehtun!

- Entitäten in Hibernate / JPA repräsentieren jeweils aktuellen Stand
- Aber was wenn man eine Änderungshistorie braucht?
- Dann verwendet man Hibernate Envers, und zwar so:

```
@Entity
```

```
@Audited
```

```
public class Kunde {
```

```
    // was hier auch sonst stehen würde...
```

```
}
```

- Na, erstmal noch mehr Arbeit:
 - EventListener für Hibernate-Events müssen konfiguriert werden
 - post-insert, -update, -delete
 - pre-collection-remove, -delete, -recreate
 - Weitere optionale Einstellungen für Envers
- Dann noch ein paar Tabellen:
 - Zu jeder überwachten Klasse/Entität wird eine Tabelle `<NameDerEntität>_AUD` erzeugt
 - Eine weitere Tabelle `REVINFO` wird erzeugt

- Naja,
 - jedes persist(), merge(), remove() einer überwachten Entität
 - erzeugt (neben dem Aktualisieren der ursprünglichen Tabelle) automatisch einen Eintrag in der *_AUD- und der REVINFO-Tabelle

REVINFO	REV	REVTSTMP
	43	123454566
	42	123343434

Kunde	ID	Name	Vorname
	1	Schmidt	Hans

Kunde_AUD	REV	REVTYPE	ID	Name	Vorname
	43	MOD	1	Schmidt	Hans
	42	MOD	1	Meyer	Hans

```

AuditReader reader = AuditReaderFactory.get(entityManager);
AuditQuery query = reader.createQuery();

// alle Entitäten in Version 42 vom Typ Kunde
query.forEntitiesAtRevision(Kunde.class, 42);

// oder alle Versionen der Entität Kunde
query.forRevisionsOfEntity(Kunde.class, false, true)
    .add(AuditEntity.property("vorname").eq("Hans"));

List result = query.getResultList();
    
```

- Das kommt drauf an...
 - Nicht alle Referenztypen werden unterstützt (Bags)
 - Hibernate 3.3 wird nur über Backport-Releases unterstützt
 - Hibernate 3.5 enthält Envers als Core-Modul
 - Zugriff auf Historie nicht immer intuitiv
 - Beim „Löschen“ einer Entität werden alle Properties der AUD-Tabelle auf Null gesetzt (natürlich abgesehen von ID, REV und REVTYPE)
 - z.B. Suche nach Historie für alle Einträge mit Vorname „Hans“ wird keine Löschereignisse liefern...aber auch das ist lösbar...
 - *_AUD-Tabellen sind nur über „native“ Queries zugreifbar (es gibt ja keine Klassen zu den *_AUD-Tabellen)

Alles weitere unter:

jboss.org/envers