



Hibernate Search

Orientation in Objects GmbH

Weinheimer Str. 68
68309 Mannheim

www.oio.de
info@oio.de

Version: 1.0

Ihre Speaker



Serge Ndong



Andreas Spall

- Einführung
- Vergleich SQL Suche, Volltextsuche
- Lucene
- Hibernate Search Architektur
- Konfiguration
- Mapping
- Queries
- Ausblick

try to find something
by looking or otherwise
seeking carefully
and thoroughly

Approximation (oder **Synonym**)
,hibernate‘ statt hibernate

Wildcard und **Wortsuche**

%hibernate%

brauchbaren Ergebnisse zuerst

Relevanz

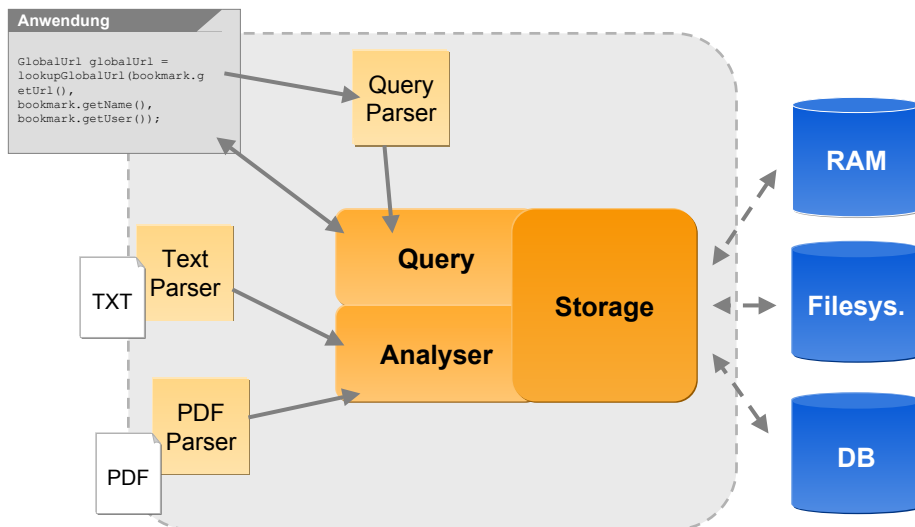
,Persistenz‘ nah zu ,**Hibernate**‘
Bedeutungsnahe

Spaltengewichtung

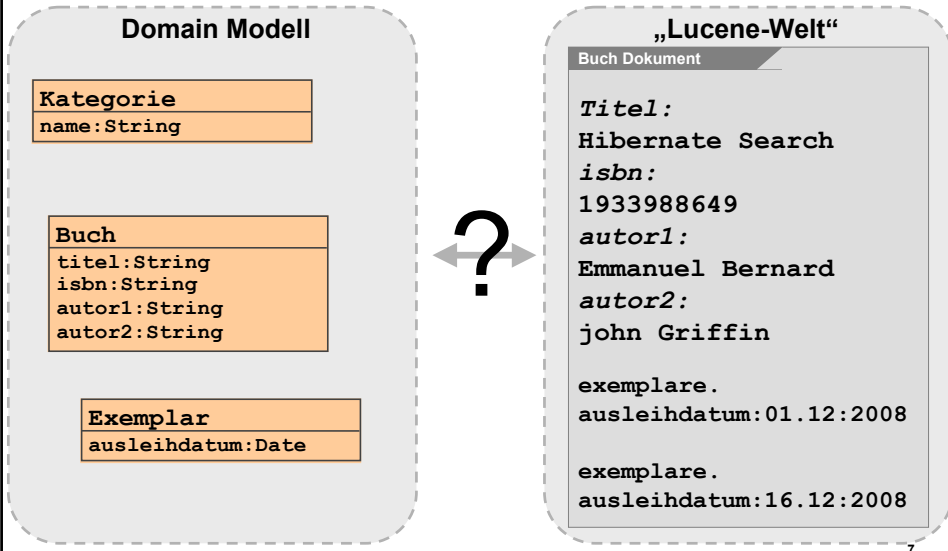
Suchen auf **mehreren Spalten**

„a book about persisting Objects
with ybernate from Wang“

5



6



Hibernate Eventsystem
Synchron oder Asynchron
Transparente Indizierung
Objektstruktur in Indexstruktur Transfer

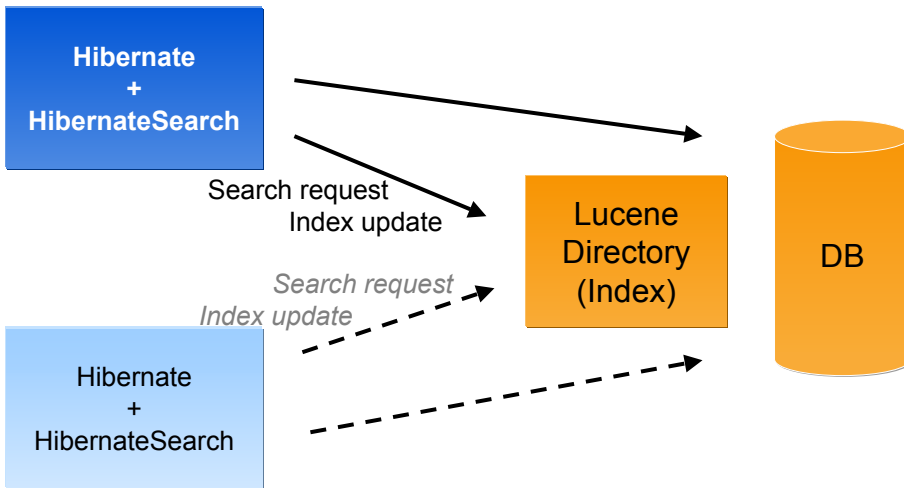
verbindet Lucene mit Hibernate

Hibernate Plattform

Licence - LGPL

verwendet Hibernate Core

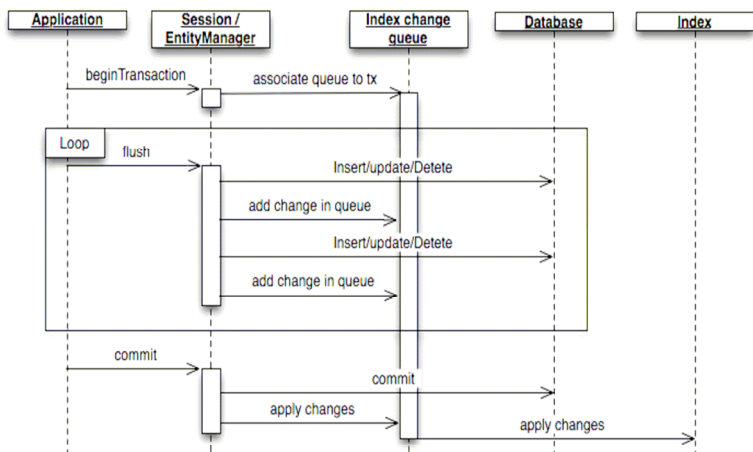
Architektur -Backend



Quelle: http://www.hibernate.org/hib_docs/search/reference/en/html/search-architecture.html

9

Backend



Quelle: *Hibernate Search in Action*(eBook)

10

- Configuration by Exception
 - 80% default Konfiguration

hibernate.properties

```
#Hibernate Search Configuration
hibernate.search.default.indexBase /users/application/indexes
```

hibernate.cfg.xml

```
<property name=„ hibernate.search.default.indexBase“>
    /users/application/indexes
</property>
```

11

Hibernate Search

© 2008 Orientation in Objects GmbH

```
@Indexed(index="buechererverwaltung")
@Entity@Table(name = "Buecher")
public class Buch {

    @DocumentId
    @Id @Column(name = "buch_id")
    private Integer id;

    @Field
    private String titel;

    @Field(store = Store.YES, index = Index.UN_TOKENIZED)
    private String isbn;

    @IndexedEmbedded
    @ManyToOne @JoinColumn(name = "verlag_FK")
    private Verlag herausgeber;

    @ContainedIn
    @OneToMany(mappedBy = "buch") @JoinColumn(name = "Buch_FK")
    private Set<Exemplar> exemplare = new HashSet<Exemplar>();
```

12

Hibernate Search

© 2008 Orientation in Objects GmbH

Field Bridge

```
@Field
private String titel;
```

- Built-in
- Konvertierung vom Objekttyp zum String
- Vergleich

Index Strategien

```
@Field
@Field(name = "isbn_sort", index = Index.UN_TOKENIZED,
store = Store.YES)
private String isbn;
```

- Speicherung von Daten im Index
- Eine Eigenschaft mehrfach Indizieren

13

Analyser

```
@Field
@Analyzer(impl=PhoneticAnalyzer.class)
private String titel;
```

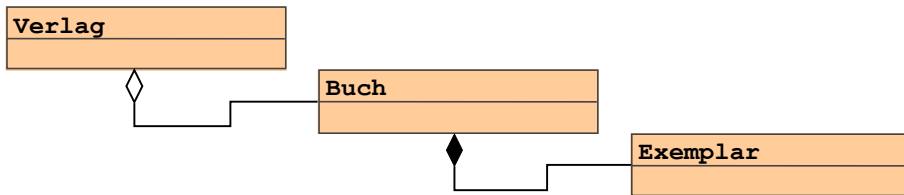
- Text in Wörter zerstückeln (auf Tokenized Felder)
- Default Analyzer Konfigurierbar
- auch auf Entität- und Feldebene einsetzbar

Boost Factor

```
@Field
@Boost(1.5f)
private String oioTags;
```

- default ist 1.0
- Beeinflusst den Document Rank
- Wert für Feld = Entity Wert * Feld Wert

14



Relation

```
@IndexedEmbedded
@ManyToOne @JoinColumn(name = "verlag_FK")
private Verlag herausgeber;

@ContainedIn
@OneToMany(mappedBy = "buch") @JoinColumn(name = "Buch_FK")
private Set<Exemplar> exemplare = new HashSet<Exemplar>();
```

15

- Abfrage ähnlich wie JPA und Criteria Query
- Objekte werden zurückgegeben, keine Dokumente
 - kein Code zum Konvertieren
- Objekte aus dem Persistenzkontext
 - “Hibernate managed“ Objekten werden zurückgegeben
- Benutzt die `org.hibernate.Query /javax.persistence.query`
 - Unterstützung von Paging
- Query auf abhängige Objekten
 - Join ähnliche Queries

16

1. Lucene Query

2. Hibernate Search wrappen

```
MultiFieldQueryParser parser;
parser = new MultiFieldQueryParser(
    new String[] { "title", "descr"},
    new StandardAnalyzer());

org.apache.lucene.search.Query query = null;
query = parser.parse("o wie OIO");
```

```
// org.hibernate.search.jpa.FullTextEntityManager
// org.hibernate.search.jpa.Search
FullTextEntityManager fullTextEM;
fullTextEM = Search.createFullTextEntityManager(entityMang);

// org.hibernate.search.jpa.FullTextQueryFullTextQuery
FullTextQueryFullTextQuery query;
query = fullTextEM.createFullTextQuery(query, Book.class);
```

17

3. Params

4. Happiness

```
query.setFirstResult(15);
query.setMaxResults(10);

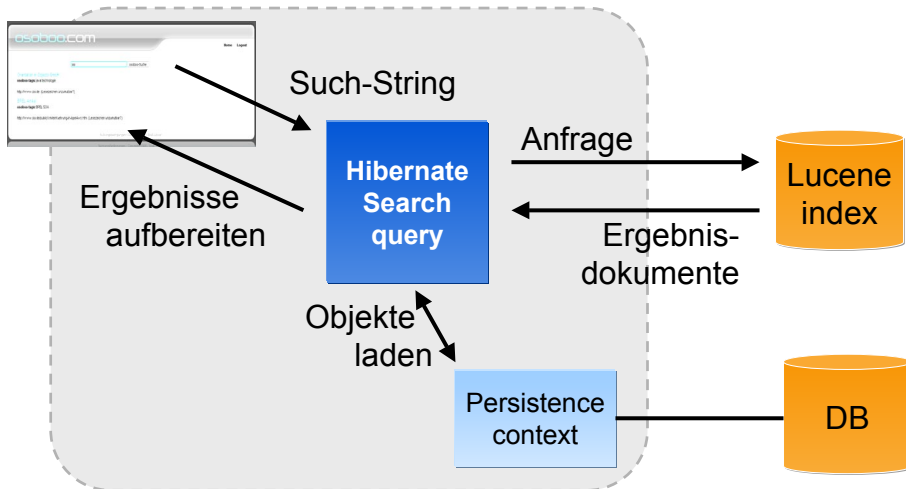
// gibt dann nur die document_id zurück
// welches meist auch die Id ist
query.setProjection(FullTextQuery.ID);
```

```
List<Object[]> results = hibQuery.getResultList();

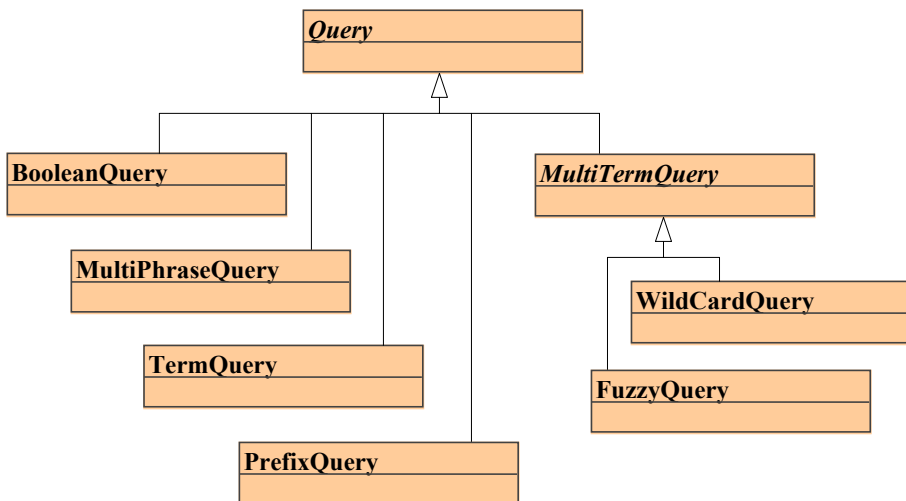
List<BookDTO> booksView = new ArrayList<BookDTO>();
for (Object[] result : results) {
    String id = (String) result[0];
    booksView.add(findAsDTO(id));
}
return booksView;
```

18

Query Ausführungsschritte



org.apache.lucene.search.Query



- Sortierung bei Felden statt bei der Relevanz
 - Index.UN_TOKENIZED oder INDEX_NO_NORMS
- Natives Lucene
- Manuelle Indizierung
- Index Sharding

Proktion

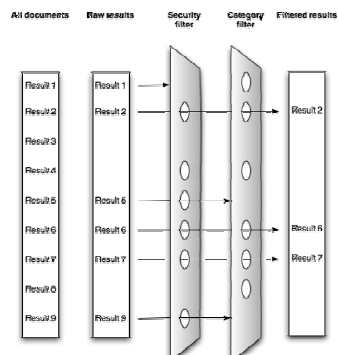
properties statt
„managed“ Objekte

metadata

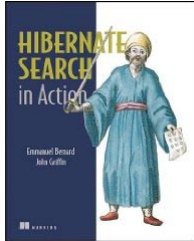
SCORE
DOCUMENT_ID
BOOST
...

Filter

Security
Kategorien



Quelle: Hibernate Search in Action



- Hibernate Search in Action
 - **Sprache: Englisch**
broschiert - 450 Seiten - Manning
publications
 - **Erscheinungsdatum: 28. Dezember 2008**
ISBN: 1933988649

23

- Referenz Guide
 - http://www.hibernate.org/hib_docs/search/reference/en/html/
- Hibernate Search
 - <http://search.hibernate.org>
- Apache Lucene
 - <http://lucene.apache.org>
 - Lucene in Action - Manning

24



Vielen Dank für Ihre Aufmerksamkeit !

Orientation in Objects GmbH

Weinheimer Str. 68
68309 Mannheim

www.oio.de
info@oio.de

Version: 1.0



Fragen ?

Orientation in Objects GmbH

Weinheimer Str. 68
68309 Mannheim

www.oio.de
info@oio.de

Version: 1.0