Search Evolution – von Lucene zu Solr und ElasticSearch

Florian Hopf

@fhopf

http://www.florian-hopf.de

BED BERLIN EXPERT DAYS 2013

{join the talk}

Das Programm der BED-Con 2013 in der Übersicht

Haben Sie Fragen zum Programm? Bitte schreiben Sie eine E-Mail an uns michael.schuetz {at} bed-con (dot) org. Bei gegebenem Anlass können noch kurzfristig Änderungen im Programm erfolgen.

Donnerstag, 04. April 2013

Zeit	Hörsaal 1	Hörsaal 2	Seminarraum 1	Seminarraum 2
ab 08:00	Einlass			
09:00 – 10:00	**JavaScript + Java EE = ♥** Kris Borchers EN	**OAuth 2.0 – Ein Standard wird erwachsen** Uwe Friedrichsen	**Search Evolution – Von Lucene zu Solr und ElasticSearch** Florian Hopf	**Offline Strategien für HTML5 Web Applikationen** Stephan Hochdörfer
10:00 – 10:15	Pause			
10:15 – 11:15	**Java EE 7, the road ahead** David Delabassee EN	**Memory Management: TP, CMS und G1 – welche GC-Strategie ist die richtige?** Tobias Frech	**Kurzvortrag-Session** **Guttenbase** Markus Dahm **DB versionieren** Niko Köbler **Togglz** Niko Köbler	**Spring Data Repositories unter der Lupe** Oliver Gierke
11:15 – 11:30	Pause			

datenbank Suchen

DB versionieren

Dein Quellcode ist versioniert im Repository abgelegt! Warum Deine **Datenbank** nicht?Jederzeit einen beliebigen Stand der **Datenbank** in der Entwickler-, Test- oder Produktions-Umgebung wiederherstellen? Machs doch einfach! Die Flyway-Bibliothek lässt sich nahtlos in jede Java-Anwendung und in den Build für agiles Continuous Delivery integrieren. Es war noch nie so einfach!

Handling humongous data with NoSQL/MongoDB

Der Umgang mit schnell wachsenden Datenmengen, sich ändernden Strukturen sowie dem Wunsch nach Skalierbarkeit stellt herkömmliche RDBMS System vor neue Herausforderungen. Eine adäquate Lösung hierfür bieten mittlerweile NoSQL **Datenbanken**. MongoDB wird als prominenter Vertreter der Dokumentorientierten **Datenbanken** detailliert vorgestellt. Neben des Basics werden u.a. Sharding, Replica Sets, Map/Reduce und das Schema Design aufgegriffen.

Guttenbase

Aus vielerlei Gründen müssen oft komplette **Datenbanken** kopiert oder migriert werden.Z.B., um lokal entwickeln zu können oder damit eine separate Anwendung mit den selben Datenarbeiten kann. Schwierig wird eine Migration insbesondere zwischen verschiedenen RDBMS.Bisherige Werkzeuge sind für diese Aufgaben oft unzureichend:Eine Lösung bietet das Framework “GuttenBase”, mit dem man Datenmigrationen programmieren kann.Dies ist ein wesentlicher Unterschied zu bestehenden Werkzeugen

Logdateien live und in Farbe

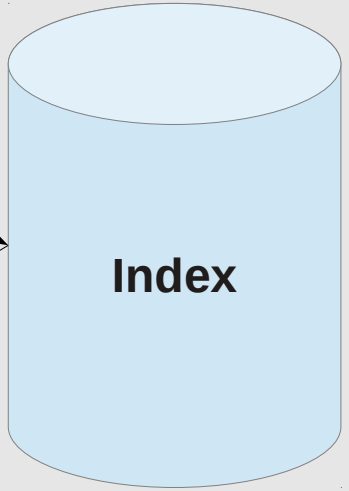
Wenn Log-Informationen in Dateien landen ist es meist ein Datenfriedhof. Spätestens bei der Fehlersuche in der Produktion zeigen sich die Grenzen, wenn die Logdateien über verschiedene Server verstreut, die Dateien groß sind, und der Weg über das Operating für den Zugriff lang ist.Ein Logserver bringt hier Ordnung: Historische Daten können in einer (No)SQL-**Datenbank** gespeichert und gefunden werden, Events können live und in Farbe am Bildschirm mitverfolgt werden, der Zugriff ist nach Anmeldung

Haskell aus einer Java-Enterprise Perspektive

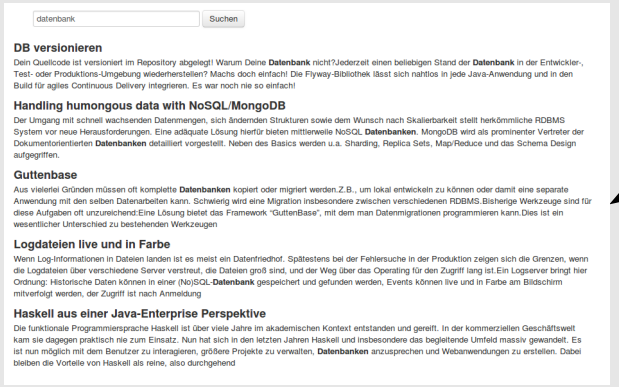
Die funktionale Programmiersprache Haskell ist über viele Jahre im akademischen Kontext entstanden und gereift. In der kommerziellen Geschäftswelt kam sie dagegen praktisch nie zum Einsatz. Nun hat sich in den letzten Jahren Haskell und insbesondere das begleitende Umfeld massiv gewandelt. Es ist nun möglich mit dem Benutzer zu interagieren, größere Projekte zu verwalten, **Datenbanken** anzusprechen und Webanwendungen zu erstellen. Dabei bleiben die Vorteile von Haskell als reine, also durchgehend

Index

Indizieren



Suchen



Index

Term	Document Id

Analyzing

http://www.flickr.com/photos/quinnanya/5196951914/

Analyzing

Such Evolution - Von Lucene zu Solr und ElasticSearch

Verteiltes Suchen mit Elasticsearch

Analyzing

Such Evolution - Von Lucene zu Solr und ElasticSearch

Verteiltes Suchen mit Elasticsearch

1. Tokenization

Term	Document Id
Such	1
Evolution	1
Von	1
Lucene	1
zu	1
Solr	1
und	1
ElasticSearch	1
Verteiltes	2
Suchen	2
mit	2
Elasticsearch	2

Analyzing

Such Evolution - Von Lucene zu Solr und ElasticSearch

Verteiltes Suchen mit Elasticsearch

1. Tokenization

2. Lowercasing

Term	Document Id
such	1
evolution	1
von	1
lucene	1
zu	1
solr	1
und	1
elasticsearch	1,2
verteiltes	2
suchen	2
mit	2

Analyzing

Such Evolution - Von Lucene zu Solr und ElasticSearch

Verteiltes Suchen mit Elasticsearch

1. Tokenization

2. Lowercasing

3. Stemming

Term	Document Id
such	1,2
evolution	1
von	1
luc	1
zu	1
solr	1
und	1
elasticsearch	1,2
verteilt	2
mit	2


Lucene

Inverted Index


Analyzer

Query Syntax

datenbank OR DB

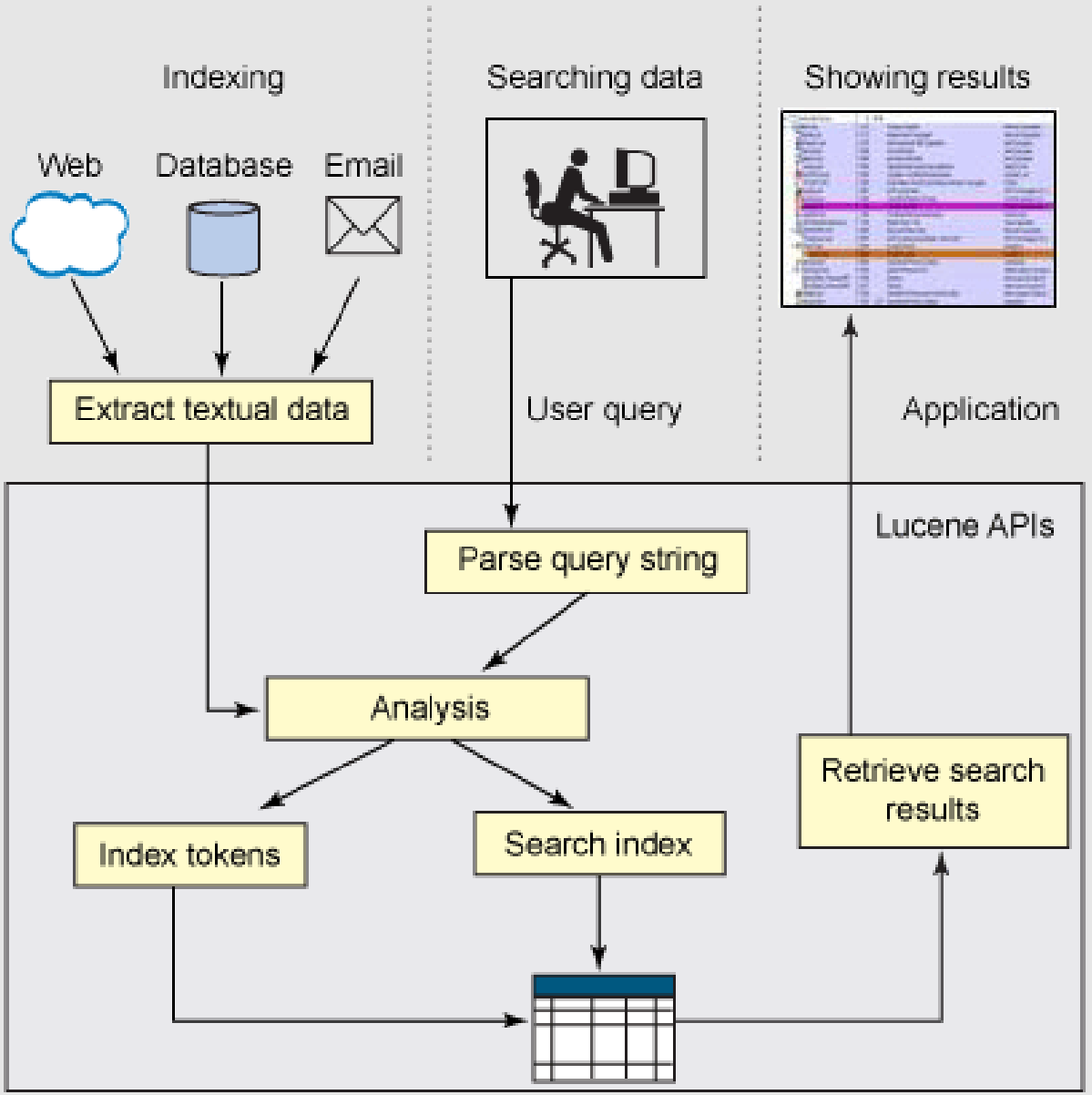
title:elasticsearch

"apache lucene"

speaker:hopp~

elastic* AND date:[20130101 TO 20130501]

Relevance



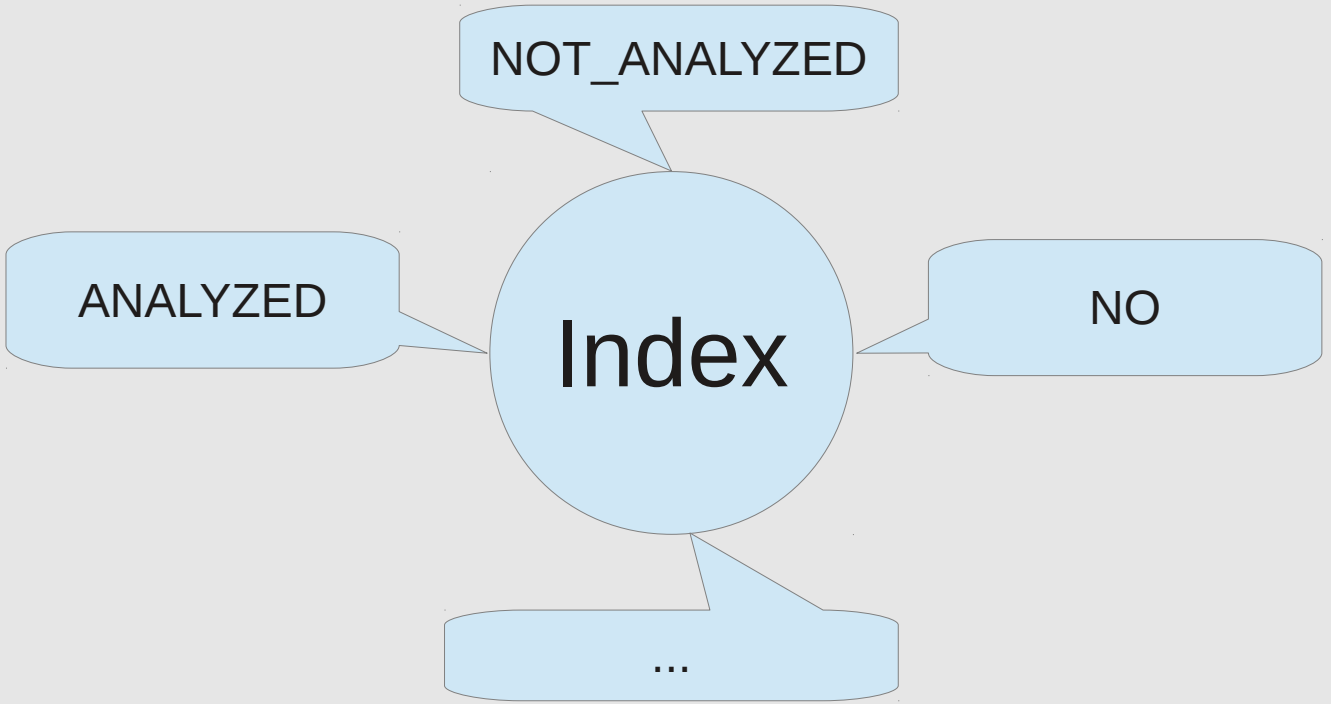
http://www.ibm.com/developerworks/java/library/os-apache-lucenesearch/

Documents

Document

title	Verteiltes Suchen mit Elasticsearch
date	20130404
speaker	Dr. Halil-Cem Gürsoy

Attributes



Attributes

Indexing

```
Document es = new Document();
es.add(new Field("title",
   "Verteiltes Suchen mit Elasticsearch",
   Field.Store.YES,
   Field.Index.ANALYZED));
es.add(new Field("date",
   "20130404",
   Field.Store.NO,
   Field.Index.ANALYZED));
es.add(new Field("speaker",
   "Dr. Halil-Cem Gürsoy",
   Field.Store.YES,
   Field.Index.ANALYZED));
```

Indexing

```
Directory dir = FSDirectory.open(
   new File("/tmp/testindex"));
IndexWriterConfig config = new IndexWriterConfig(
   Version.LUCENE_36,
   new GermanAnalyzer(Version.LUCENE_36));
IndexWriter writer = new IndexWriter(dir, config);

writer.addDocument(es);

writer.commit();
```

Searching

```
IndexReader reader = IndexReader.open(dir);
IndexSearcher searcher = new IndexSearcher(reader);
QueryParser parser = new QueryParser(
      Version.LUCENE_36,
      "title",
      new GermanAnalyzer(Version.LUCENE_36));
Query query = parser.parse("suche");

TopDocs result = searcher.search(query, 10);
assertEquals(1, result.totalHits);

int id = result.scoreDocs[0].doc;
Document doc = searcher.doc(id);
String title = doc.get("title");
assertEquals(
      "Verteiltes Suchen mit Elasticsearch",
      title);
```

datenbank Suchen

DB versionieren

Dein Quellcode ist versioniert im Repository abgelegt! Warum Deine **Datenbank** nicht?Jederzeit einen beliebigen Stand der **Datenbank** in der Entwickler-, Test- oder Produktions-Umgebung wiederherstellen? Machs doch einfach! Die Flyway-Bibliothek lässt sich nahtlos in jede Java-Anwendung und in den Build für agiles Continuous Delivery integrieren. Es war noch nie so einfach!

Handling humongous data with NoSQL/MongoDB

Der Umgang mit schnell wachsenden Datenmengen, sich ändernden Strukturen sowie dem Wunsch nach Skalierbarkeit stellt herkömmliche RDBMS System vor neue Herausforderungen. Eine adäquate Lösung hierfür bieten mittlerweile NoSQL **Datenbanken**. MongoDB wird als prominenter Vertreter der Dokumentorientierten **Datenbanken** detailliert vorgestellt. Neben des Basics werden u.a. Sharding, Replica Sets, Map/Reduce und das Schema Design aufgegriffen.

Guttenbase

Aus vielerlei Gründen müssen oft komplette **Datenbanken** kopiert oder migriert werden.Z.B., um lokal entwickeln zu können oder damit eine separate Anwendung mit den selben Datenarbeiten kann. Schwierig wird eine Migration insbesondere zwischen verschiedenen RDBMS.Bisherige Werkzeuge sind für diese Aufgaben oft unzureichend:Eine Lösung bietet das Framework “GuttenBase”, mit dem man Datenmigrationen programmieren kann.Dies ist ein wesentlicher Unterschied zu bestehenden Werkzeugen

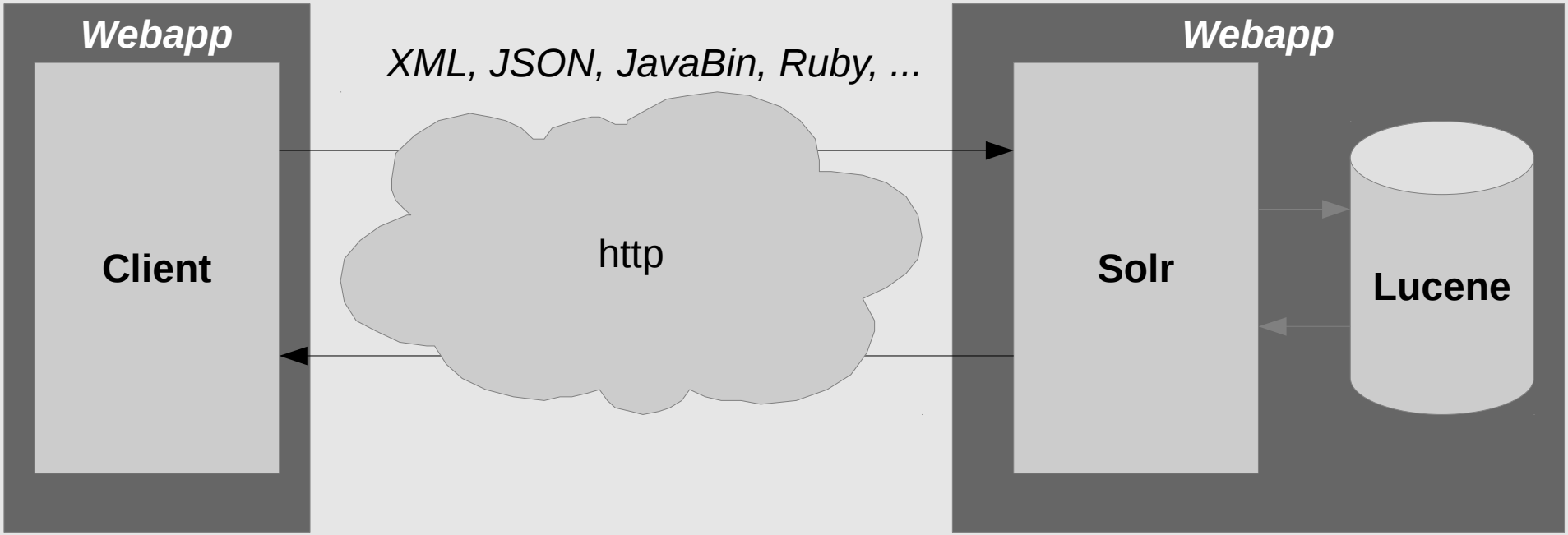
Logdateien live und in Farbe

Wenn Log-Informationen in Dateien landen ist es meist ein Datenfriedhof. Spätestens bei der Fehlersuche in der Produktion zeigen sich die Grenzen, wenn die Logdateien über verschiedene Server verstreut, die Dateien groß sind, und der Weg über das Operating für den Zugriff lang ist.Ein Logserver bringt hier Ordnung: Historische Daten können in einer (No)SQL-**Datenbank** gespeichert und gefunden werden, Events können live und in Farbe am Bildschirm mitverfolgt werden, der Zugriff ist nach Anmeldung

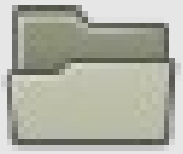
Haskell aus einer Java-Enterprise Perspektive

Die funktionale Programmiersprache Haskell ist über viele Jahre im akademischen Kontext entstanden und gereift. In der kommerziellen Geschäftswelt kam sie dagegen praktisch nie zum Einsatz. Nun hat sich in den letzten Jahren Haskell und insbesondere das begleitende Umfeld massiv gewandelt. Es ist nun möglich mit dem Benutzer zu interagieren, größere Projekte zu verwalten, **Datenbanken** anzusprechen und Webanwendungen zu erstellen. Dabei bleiben die Vorteile von Haskell als reine, also durchgehend

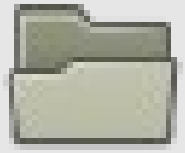

Apache
Solr


Webapp
Client
XML, JSON, JavaBin, Ruby, ...
http
Webapp
Solr
Lucene

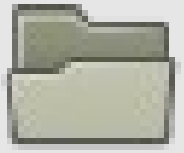
Config



Solr Home



conf

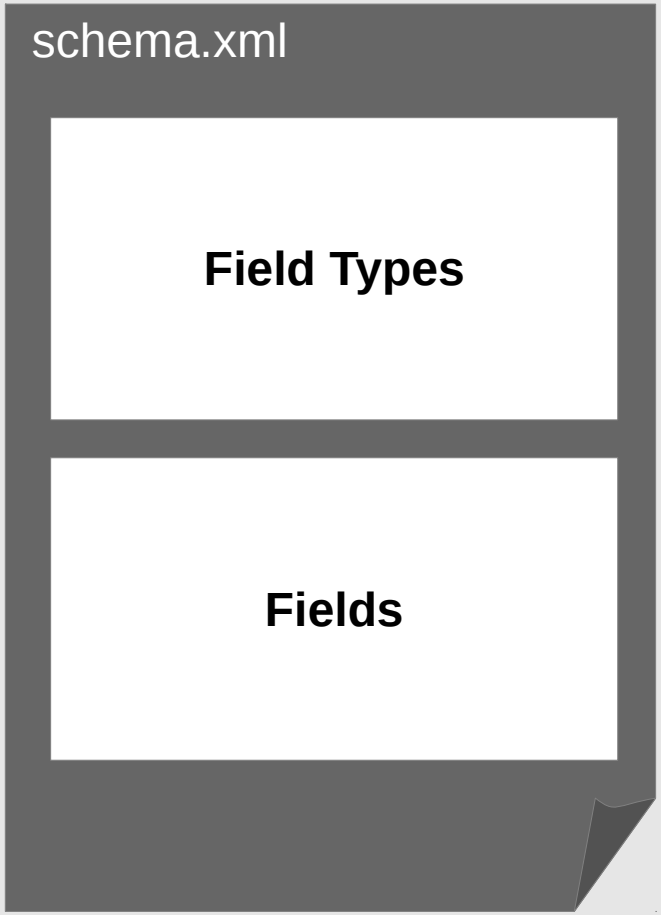


data

schema.xml

solr-config.xml

Schema



Schema

```
<fieldType name="text_de" class="solr.TextField">
   <analyzer>
      <tokenizer
         class="solr.StandardTokenizerFactory"/>
      <filter
         class="solr.LowerCaseFilterFactory"/>
      <filter
         class="solr.GermanLightStemFilterFactory"/>
   </analyzer>
</fieldType>
```

Schema

```
<fields>
   <field name="title" type="text_de"
      indexed="true" stored="true"/>
   <field name="speaker" type="string"
      indexed="true" stored="true"
      multiValued="true"/>
   <field name="speaker_search" type="text_ws"
      indexed="true" stored="false"
      multiValued="true"/>
      [...]
</fields>

<copyField source="speaker" dest="speaker_search"/>
```

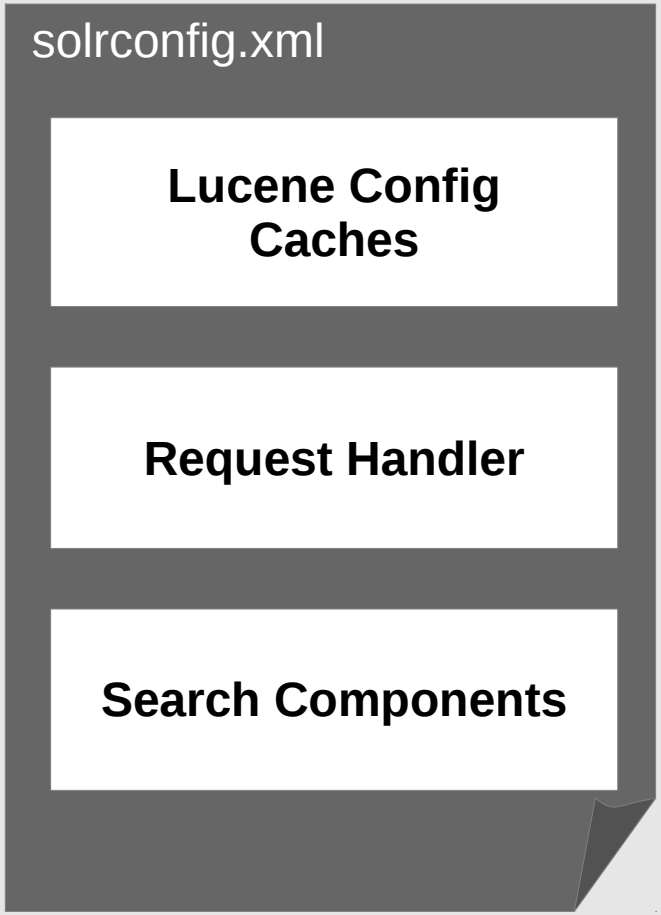
Indexing

```
SolrInputDocument document =
            new SolrInputDocument();
document.addField("path",
            "/tmp/foo");
document.addField("title",
            "Verteiltes Suchen mit Elasticsearch");
document.addField("speaker",
            "Dr. Halil-Cem Gürsoy");

SolrServer server =
   new HttpSolrServer("http://localhost:8080");

server.add(document);
server.commit();
```

Solrconfig


solrconfig.xml
Lucene Config
Caches
Request Handler
Search Components

Solrconfig

```
<requestHandler name="/bedcon"
   class="solr.SearchHandler">
   <lst name="defaults">
      <int name="rows">10</int>
      <str name="q.op">AND</str>
      <str name="q.alt">*:*</str>
      <str name="defType">edismax</str>
      <str name="qf">
         content
         title^1.5
         speaker_search
      </str>
   </lst>
</requestHandler>
```

Searching

```
SolrQuery solrQuery = new SolrQuery("suche");
solrQuery.setQueryType("/bedcon");

QueryResponse response = server.query(solrQuery);
assertEquals(1, response.getResults().size());

SolrDocument result = response.getResults().get(0);
assertEquals("Verteiltes Suchen mit Elasticsearch",
             result.get("title"));
assertEquals("Dr. Halil-Cem Gürsoy",
             result.getFirstValue("speaker"));
```

Nach Datum sortieren

build Suchen

Speaker:

- Andreas Lüdeke (1)
- Claus Ibsen (1)
- David Delabassee (1)
- Jan Jongboom (1)
- Manuel Blechschmidt (1)
- Niko Köbler (1)
- Rene Groeschke (1)
- René Gröschke (1)
- Robert Schuster (1)

Datum:

- 2012 (5)
- 2013 (4)

Gradle wird den *Build* schon schaukeln ...

Gradle ist ein innovatives Open-Source-*Build*- und Automatisierungstool. Es verbindet die Vorzüge ...

How to *build* a recommender system based on Mahout and Java EE ...

Speaker: Manuel Blechschmidt, Language: EnglishFolien: How to *build* a recommender system based on ...

Gradle, der neue Stern am Himmel der Open-Source-*Build*-Systeme ...

Gradle ist der neue Stern am Himmel der Open-Source-*Build*-Systeme. In dieser Session wird anhand ...

DB versionieren

doch einfach! Die Flyway-Bibliothek lässt sich nahtlos in jede Java-Anwendung und in den *Build* für ...

Java EE 7, the road ahead

language. Long awaited Batch Processing API and Caching API are also getting added to *build* applications ...

AngularJS made easy with Yeoman

implementierung auch eingefleischte JavaScript-Phobiker beeindruckt. Zusammen mit Yeoman als JS *Build* Tool haben ...

Enterprise Integration Patterns and DSL with Apache Camel

Apache Camel, a very popular integration framework, *builds* on the principles of the EIPs ...

Maven Packaging Plugin – Creating distribution packages for Java software artifacts

IPK-based GNU/Linux-distributions and integrates well with *build* servers such as Jenkins.The talk will ...

JavaScript in the Cloud

it ourselves – *build* an IDE in the browser: Cloud9 IDE.In this talk I will describe the vision and ...

Faceting

```
...

solrQuery.setFacet(true);
solrQuery.addFacetField("speaker");

QueryResponse response = server.query(solrQuery);
List<FacetField.Count> speakerFacet =
     response.getFacetField("speaker").getValues();
assertEquals(1, speakerFacet.get(0).getCount());
assertEquals("Dr. Halil-Cem Gürsoy",
     speakerFacet.get(0).getName());
```

elasticsearch.

Indexing

```
curl -XPOST
   'http://localhost:9200/bedcon/talk/' -d '{
    "speaker" :
        "Dr. Halil-Cem Gürsoy",
    "date"    :
        "2013-04-04T16:00:00",
    "title"   :
        "Verteiltes Suchen mit Elasticsearch"
}'

{"ok":true,"_index":"bedcon","_type":"talk",
"_id":"CeltdivQRGSvLY_dBZv1jw","_version":1}
```

Mapping

```
curl -XPUT
   'http://host/bedcon/talk/_mapping' -d '{
   "talk" : {
      "properties" : {
      "title" : {
            "type" : "string",
            "analyzer" : "german"
         }
      }
   }
}'
```

Searching

```
curl -XGET
   'http://host/bedcon/talk/_search?q=elasticsearch'
{...},
"hits":{"total":1,"max_score":0.054244425,
   "hits":[{
      ...,
      "_score":0.054244425,
      "_source" : {
        "speaker" :
           "Dr. Halil-Cem Gürsoy",
        "date" :
           "2013-04-04T16:00:00",
        "title":
           "Verteiltes Suchen mit Elasticsearch"
      }
   }
}
```

Searching

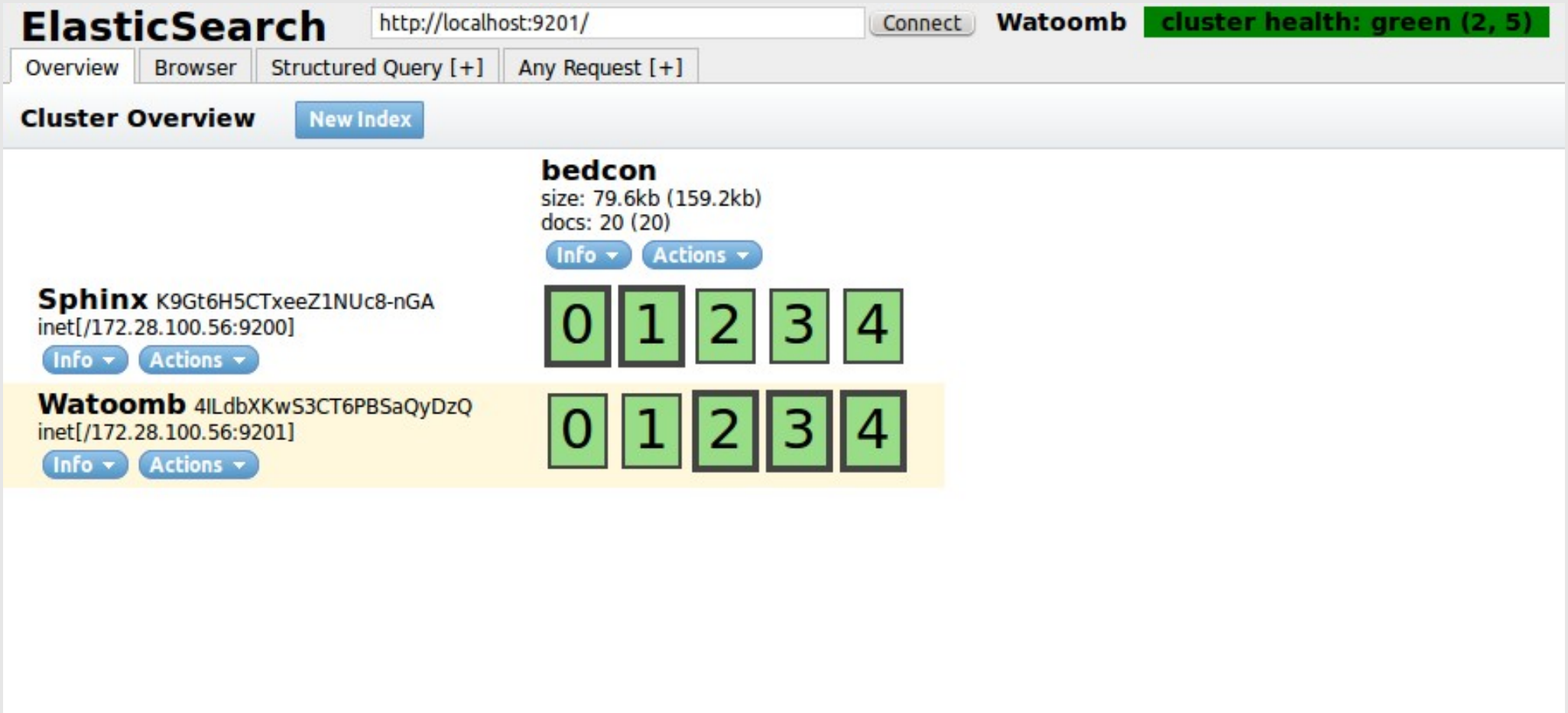
```
curl -XGET
'http://localhost:9200/bedcon/talk/_search' -d '{
   "query" : {
      "query_string" : {"query" : "elasticsearch"}
   },
   "facets" : {
      "tags" : {
         "terms" : {"field" : "speaker"}
      }
   }
}'
```

Searching

```
SearchResponse response =
   esClient.prepareSearch("bedcon")
   .addFacet(
      FacetBuilders.termsFacet("speaker")
                   .field("speaker"))
   .setQuery(
      QueryBuilders.queryString("elasticsearch"))
   .execute().actionGet();

assertEquals(1, response.getHits().getTotalHits());
```

Verteilung


ElasticSearch
http://localhost:9201/
Connect
Watoomb
cluster health: green (2, 5)
Overview
Browser
Structured Query [+]
Any Request [+]
Cluster Overview
New Index
bedcon
size: 79.6kb (159.2kb)
docs: 20 (20)
Info
Actions
Sphinx K9Gt6H5CTxeeZ1NUc8-nGA
inet[/172.28.100.56:9200]
Info
Actions
0 1 2 3 4
Watoomb 4lLdbXKwS3CT6PBSaQyDzQ
inet[/172.28.100.56:9201]
Info
Actions
0 1 2 3 4

Verteilung


ElasticSearch
http://localhost:9201/
Connect
Watoomb
cluster health: green (3, 5)
Overview
Browser
Structured Query [+]
Any Request [+]
Cluster Overview
New Index
bedcon
size: 79.6kb (159.2kb)
docs: 20 (20)
Info
Actions
Dweller-in-Darkness 27iedtOQSjWwBnCl_ZRsDw
inet[/172.28.100.56:9202]
Info
Actions
0 1 2
Sphinx K9Gt6H5CTxeeZ1NUc8-nGA
inet[/172.28.100.56:9200]
Info
Actions
0 1 3 4
Watoomb 4ILdbXKwS3CT6PBSaQyDzQ
inet[/172.28.100.56:9201]
Info
Actions
2 3 4


http://lucene.apache.org
http://lucene.apache.org/solr/
http://elasticsearch.org
https://github.com/fhopf/lucene-solr-talk
@fhopf
mail@florian-hopf.de
http://blog.florian-hopf.de

Images

- http://www.morguefile.com/archive/display/3470
- http://www.flickr.com/photos/quinnanya/5196951914/
  Quinn Dombrowski
- http://www.morguefile.com/archive/display/695239
- http://www.morguefile.com/archive/display/93433
- http://www.morguefile.com/archive/display/811746
- http://www.morguefile.com/archive/display/12965
- http://www.morguefile.com/archive/display/181488