

Solr Enterprise Search Server 구축하기

중소기업을 위한 오픈소스 기반 검색기 개발방법

2015. 10.

주식회사 쿼리젯 김지훈

01 발표자 소개

02 검색엔진 및 Solr 개요

03 설치 및 실행

04 텍스트 분석

05 문서 색인

06 문서 검색

07 고급 검색 기능

08 스케일 아웃 전략

09 식사 및 휴식

10 실습



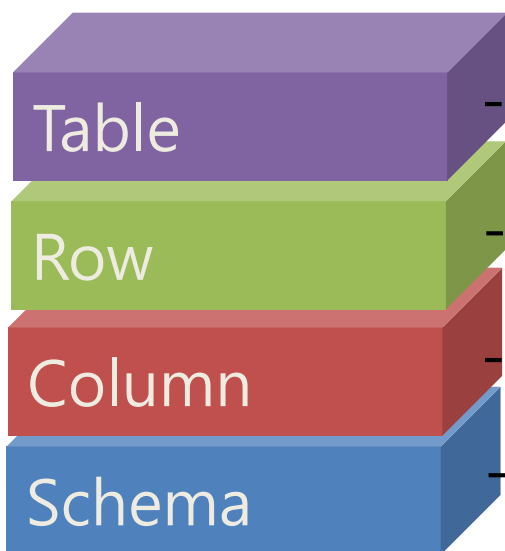
01 발표자 소개



02 검색엔진 개요 및 Solr 개요

데이터 베이스와는 무엇이 다른가?

RDBMS



Search Engine



트랜잭션 여부

01

ACID(원자성, 일관성, 독립성, 지속성)를
보장하는가?

인덱스 Term의 생성

02

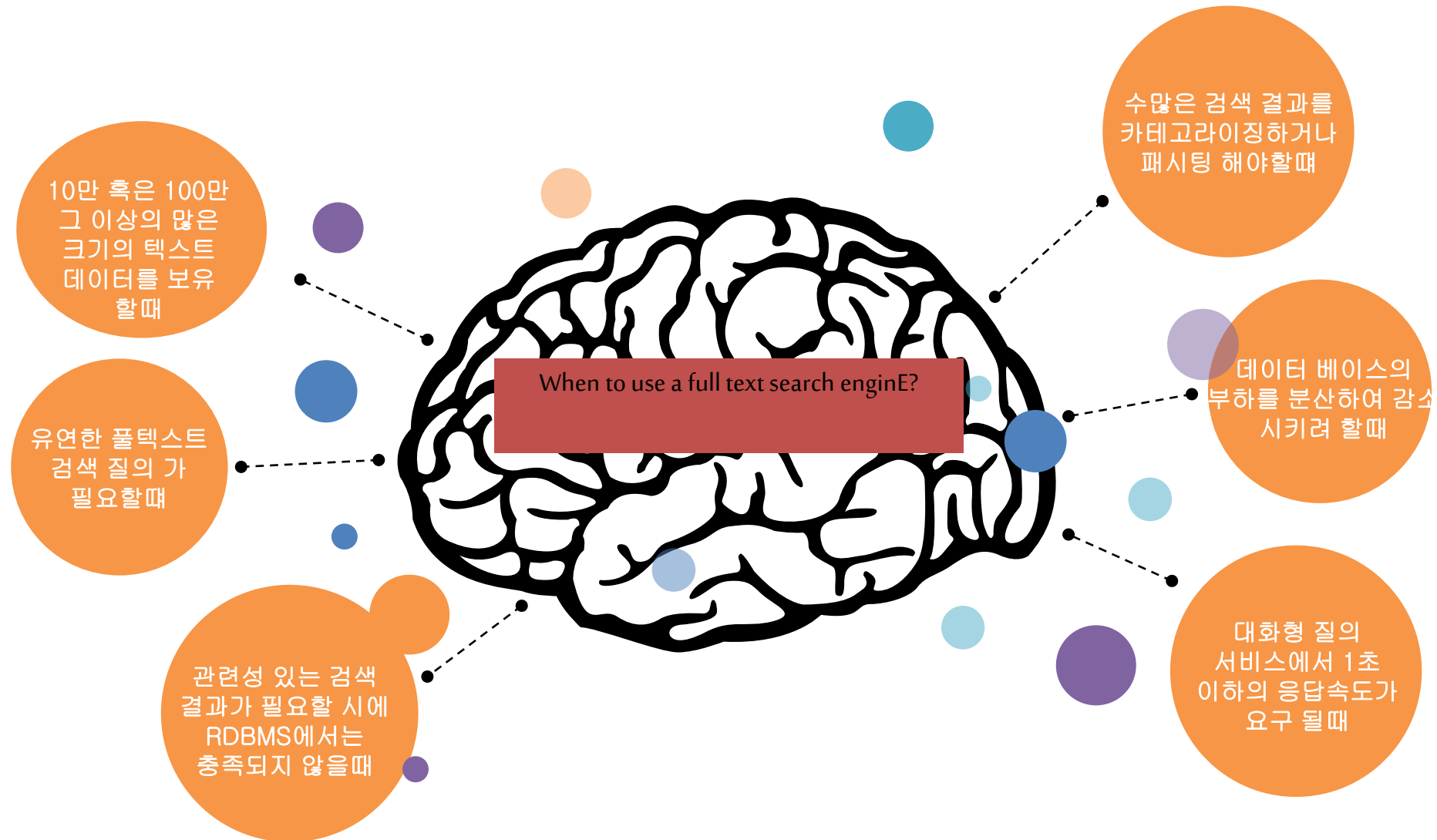
자유롭게 Term을 조직화 할수 있는가?

랭킹 알고리즘

03

다양한 랭킹알고리즘을 적용 가능한가?

어떨 때 검색엔진을 사용할까?



버티컬 검색 서비스 구축 사이클



SOLR 개요



2004년 Yonik Seeley 가 CNET Networks에서 사내용으로 사용하기 위해 루씬을 기반으로 최초 개발



2006년 아파치 소프트웨어 재단에 기부

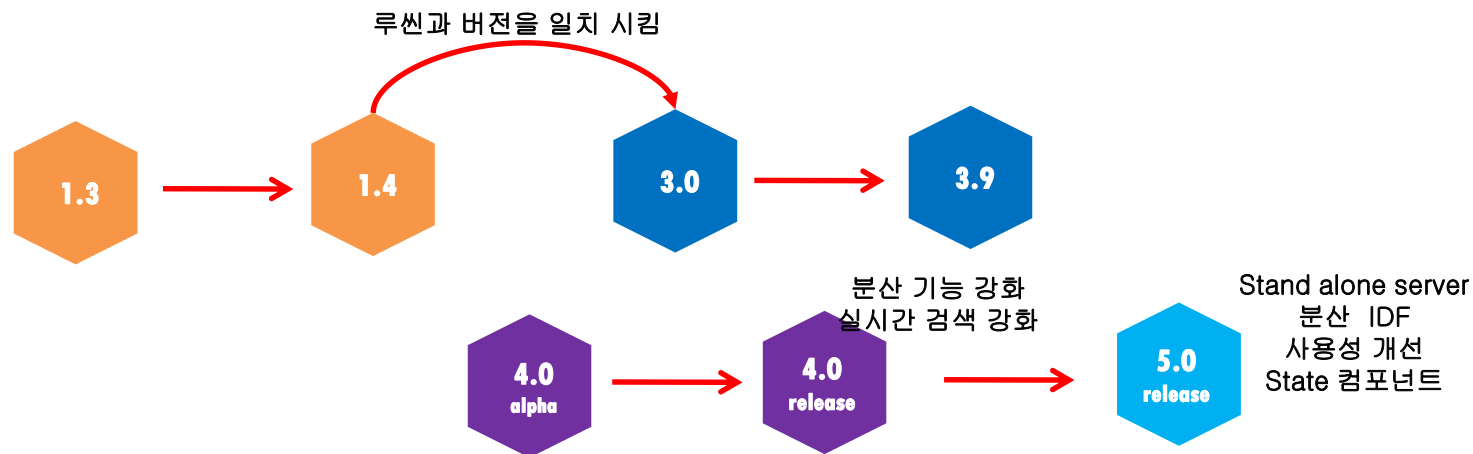


2007년 인큐베이팅을 거쳐 아파치 최상위 레벨 프로젝트로 등록



2009년 Grant Ingersoll 과 Erik Hatcher 함께 Lucid Works라는 회사를 설립하여 상용 및 교육 지원
(LucidWorks 주도로 개발중)

버전의 변화



주목해야 할 SOLR 의 배다른 형제 ElasticSearch

분산 검색 및 실시간 검색의 탁월한 성능과 기능을 보유하지만, **SolrCloud** 이후에 차별점이 점차 사라지고 있음.

차이점 요약





03 설치 및 실행

다운로드 및 설치

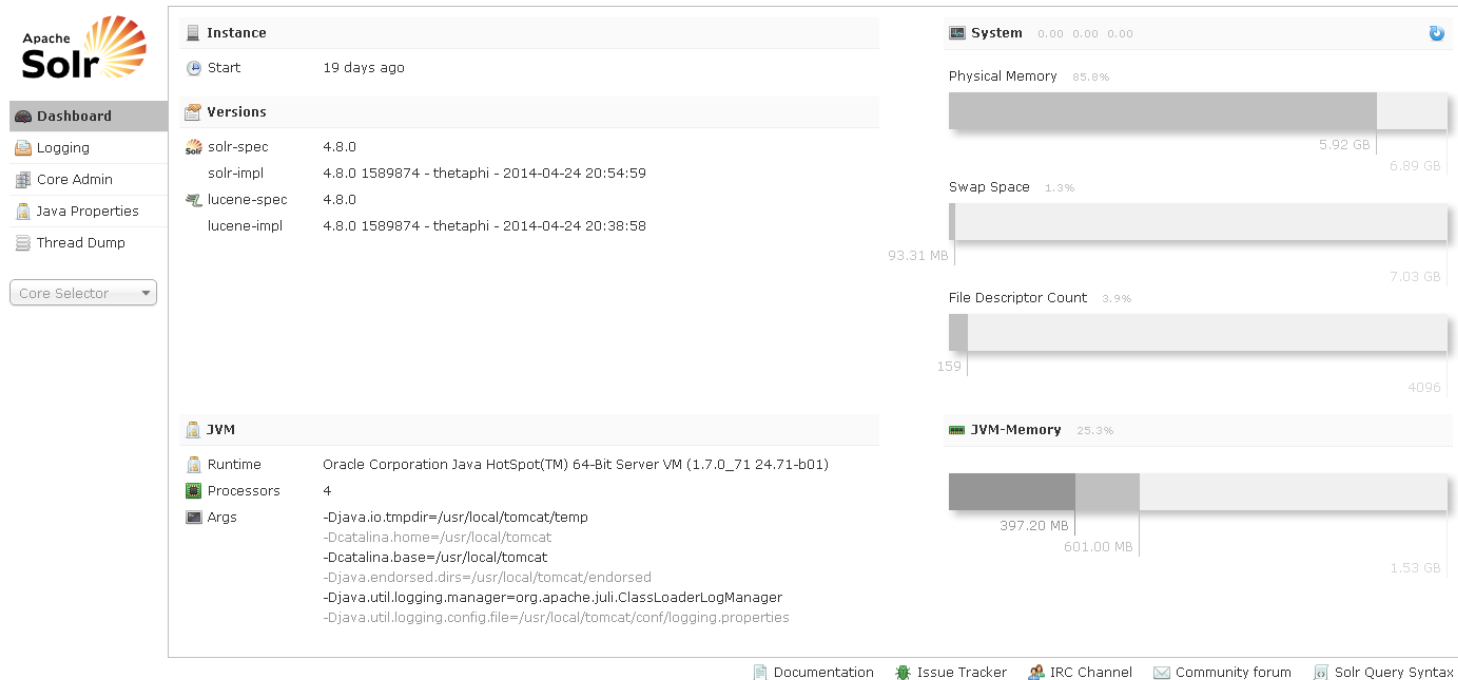
<http://apache.tt.co.kr/lucene/solr/5.3.1/>

	Parent Directory	-	-
	changes/	23-Sep-2015 20:31	-
	solr-5.3.1-src.tgz	17-Sep-2015 06:04	37M
	solr-5.3.1.tgz	17-Sep-2015 06:04	129M
	solr-5.3.1.zip	17-Sep-2015 06:04	136M

JAVA 7 이상 설치 확인

적당한 위치에 압축을 풀면 준비 완료됨.

실행 및 Admin UI 살펴보기





04 텍스트 분석

Schema.xml 설정

1) 사용할 필드 타입 정의

```
<fieldType name="string" class="solr.StrField" sortMissingLast="true" />
<fieldType name="boolean" class="solr.BoolField" sortMissingLast="true"/>
<fieldType name="int" class="solr.TrieIntField" precisionStep="0" positionIncrementGap="0"/>
<fieldType name="float" class="solr.TrieFloatField" precisionStep="0" positionIncrementGap="0"/>
<fieldType name="long" class="solr.TrieLongField" precisionStep="0" positionIncrementGap="0"/>
<fieldType name="double" class="solr.TrieDoubleField" precisionStep="0" positionIncrementGap="0"/>
```

2) 사용할 필드 정의

```
<fields>
  <field name="Shop_No" type="string" indexed="true" stored="true" multiValued="false" required="true"/>
  <field name="Shop_Nm" type="string" indexed="true" stored="true" multiValued="false" />
  <field name="Menu_Type" type="string" indexed="true" stored="true" multiValued="true" />
  <field name="Rgn1" type="string" indexed="true" stored="true" multiValued="false" />
  <field name="Rgn2" type="string" indexed="true" stored="true" multiValued="false" />
  <field name="Rgn3" type="string" indexed="true" stored="true" multiValued="false" />
  <field name="LatLng" type="location_rpt" indexed="true" stored="true" multiValued="false" />
  <field name="_version_" type="long" indexed="true" stored="true"/>
</fields>
```

3) 유니크 키 정의

```
<uniqueKey>Shop_No</uniqueKey>
```

4) 기본 검색 필드 정의

```
<defaultSearchField>Shop_Nm</defaultSearchField>
```

5) 기본 연산자 정의

```
<solrQueryParser defaultOperator="AND"/>
```

형태소 분석

색인시점에서 적용

```
<fieldType name="text_en" class="solr.TextField" positionIncrementGap="100">
  <analyzer type="index">
    <tokenizer class="solr.StandardTokenizerFactory"/>
    <!-- in this example, we will only use synonyms at query time
    <filter class="solr.SynonymFilterFactory" synonyms="index_synonyms.txt" ignoreCase="true" expand="false"/>
    -->
    <!-- Case insensitive stop word removal.
    -->
    <filter class="solr.StopFilterFactory"
      ignoreCase="true"
      words="lang/stopwords_en.txt"
    />
    <filter class="solr.LowerCaseFilterFactory"/>
    <filter class="solr.EnglishPossessiveFilterFactory"/>
    <filter class="solr.KeywordMarkerFilterFactory" protected="protwords.txt"/>
    <!-- Optionally you may want to use this less aggressive stemmer instead of PorterStemFilterFactory:
    <filter class="solr.EnglishMinimalStemFilterFactory"/>
    -->
    <filter class="solr.PorterStemFilterFactory"/>
  </analyzer>
  <analyzer type="query">
    <tokenizer class="solr.StandardTokenizerFactory"/>
    <filter class="solr.SynonymFilterFactory" synonyms="synonyms.txt" ignoreCase="true" expand="true"/>
    <filter class="solr.StopFilterFactory"
      ignoreCase="true"
      words="lang/stopwords_en.txt"
    />
    <filter class="solr.LowerCaseFilterFactory"/>
    <filter class="solr.EnglishPossessiveFilterFactory"/>
    <filter class="solr.KeywordMarkerFilterFactory" protected="protwords.txt"/>
    <!-- Optionally you may want to use this less aggressive stemmer instead of PorterStemFilterFactory:
    <filter class="solr.EnglishMinimalStemFilterFactory"/>
    -->
    <filter class="solr.PorterStemFilterFactory"/>
  </analyzer>
</fieldType>
```

검색시점에서 적용

Lucene에서의 형태소 분석기관? Tokenizer + Filter = Analyzer

Tokenizer는 무조건 1개씩 결합 할 수 있다.

Filter는 여러 개를 결합 가능하다.

Filter는 순서가 중요하다.

Tokenizer로 나누어진 것을 Token이라 하고 Filter를 거쳐서 최종적으로 색인시에 사용하는 것을 Term이라 한다.

순서대로 텍스트 토큰이 전달

solrconfig.xml 설정

캐쉬 설정

기타 설정들



05 문서 색인

색인의 몇가지 방법

REST API

DataImporterHandler

CSV Import

XML Import

REST API 이용방법

/update 핸들러에 REST 전송

```
<doc>
<field name="Shop_No">1</field>
<field name="Shop_Nm">퀴리젯</field>
<field name="Rgn1">서울특별시</field>
<field name="Rgn2">영등포구</field>
<field name="Rgn3">영등포동</field>
<field name="LatLng">35.8754666,128.5591111</field>
</doc>
```

```
<delete><id>05991</id></delete>
<Commit/>
<Optimize/>
```

DataImportHandler 설정법

1) solrconfig.xml 에 핸들러 정의

```
<requestHandler name="/dataimport"
  class="org.apache.solr.handler.dataimport.DataImportHandler">
  <lst name="defaults">
    <str name="config">data-config.xml</str>
  </lst>
</requestHandler>
```

2) data-config.xml 를 작성

DataImportHandler 설정법

2) data-config.xml 를 작성

1) 리소스 정의

```
<dataSource name="MysqlConnect" type="JdbcDataSource" driver="com.mysql.jdbc.Driver"
    url="jdbc:mysql://localhost:3386/seminardb" user="seminar_user" password="1234"/>
```

2) Entity 정의

pk= { 엔티티의 키값 정의}

dataSource={데이터 소스 정의}

query ="Full Import 쿼리 정의"

deltaImportQuery="deltaQuery에서 리턴된 키값으로 1개 Row를 리턴하는 쿼리"

deltaQuery="마지막 색인 시간 이후의 변경된 값의 키값을 리턴"

deletedPkQuery="삭제된 Doc의 키값을 리턴하는 쿼리"

3) 데이터 베이스 컬럼과 검색엔진 필드 매칭

```
<field name="Shop_No" column="Shop_No" />
```

```
<field name="Shop_Nm" column="Shop_Nm" />
```

```
<field name="Menu_Type" column="Menu_Type" splitBy="," />
```

```
<field name="Rgn1" column="Rgn1" />
```














```
<field name="Rgn2" column="Rgn2" />
```

```
<field name="Rgn3" column="Rgn3" />
```

```
<field column="$docBoost" />
```

색인

색인의 정체

 _9.fdt	2013-08-24 오전...
 _9.fdx	2013-08-24 오전...
 _9.fnm	2013-08-24 오전...
 _9.nvd	2013-08-24 오전...
 _9.nvm	2013-08-24 오전...
 _9.si	2013-08-24 오전...
 _9_Lucene41_0.doc	2013-08-24 오전...
 _9_Lucene41_0.pos	2013-08-24 오전...
 _9_Lucene41_0.tim	2013-08-24 오전...
 _9_Lucene41_0.tip	2013-08-24 오전...
 segments.gen	2013-08-24 오전...
 segments_b	2013-08-24 오전...
 write.lock	2013-08-24 오전...

MergeFactor에 의한 세그먼트의 병합 전략

세그먼트를 병합하는 빈도와 크기를 제어

<http://blog.mikemccandless.com/2011/02/visualizing-lucenes-segment-merges.html>



06 문서 검색

검색 기본 파라미터

q={boolean 검색질의}

fq={필터링된 검색 질의}

fl = {결과값 필드}

rows= {결과 값의 개수}

start= {결과 값 시작 위치}

sort ={결과 값의 정렬 기준}

rows + start + total count = 페이징

랭킹

1) Query Time Ranking vs Index Time Ranking


- Index Time Ranking

Doc Boost 이용

- Query Time Ranking

기본은 TF - IDF + 필드별 가중치 + Function Query

예) `{!boost b=recip(ms(NOW,Reg_Date),3.16e-11,0.08,0.05)}`



07 고급 검색 기능

Facet

facet =true

facet.field = facet 대상 필드

facet.mincount = 최소 갯수

facet.limit = 출력 갯수

facet.sort = 출력 순서

회사

(주)명인 (54)
태양종합운수 (33)
(주)맨파워코.. (32)

출처

사람인 (2402)
파인드잡 (2237)
인크루트 (1675)

고용형태

정규직 (6516)
계약직 (1719)
파견직 (843)

지역

경기도 (3575)
인천광역시 (955)
대구광역시 (923)



[운전/최고의일자리/자동차부품배송](#),
태양에쓰씨엠 - 인천
2/28마감 | 운전·운송·택배·배송 | 면접시
스카우트 - 2시간 전

[운전/최고의일자리/빠레트배송/4.5](#)
(주)롯데TLS - 서울 외
1/29마감 | 운전·운송·택배·배송, 물류·유
스카우트 - 2시간 전

[\(주\)일영테크 - 생산관리부사원모집](#)
(주)일영테크 - 대구 > 달서구 대구 > 달
기술/기능/연구직 > 품질관리/생산관리
경력 - 2시간 전

[\(주\)일영테크 - 품질관리부사원모집](#)
(주)일영테크 - 대구 > 달서구 대구 > 달
기술/기능/연구직 > 품질관리/생산관리
경력 - 2시간 전

[\(주\)일영테크 - 자동차부품생산직\(로](#)
(주)일영테크 - 대구 > 달서구 대구 > 달
기술/기능/연구직 > 기계/기계설비 | 5
경력 - 2시간 전

[플라스틱 사출 생산직 채용](#)  
세원산업 - 대구
2/28마감 | 플라스틱제품제조 | 회사규정

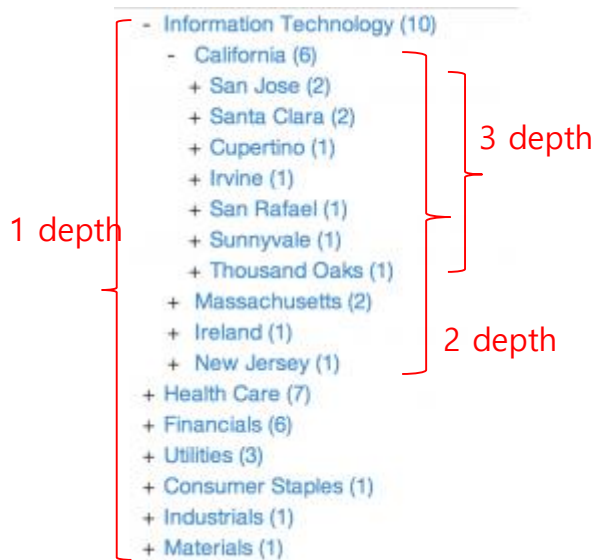
카테고리 10	+	브랜드 200
디지털/가전 139,417	<input checked="" type="checkbox"/> LG 디오스	<input type="checkbox"/> 삼성전자
주방가전 135,663	<input type="checkbox"/> 삼성 지펠	<input type="checkbox"/> LG전자
PC액세서리 840	<input type="checkbox"/> 대우 클라쎄	<input type="checkbox"/> 박씨상방
모니터주변기기 547	<input type="checkbox"/> 위니아만도 프라우드	<input type="checkbox"/> 노빌타
노트북액세서리 454	<input type="checkbox"/> 일렉트로룩스	<input type="checkbox"/> 카네이션
영상가전 414	<input type="checkbox"/> 꼬망스	<input type="checkbox"/> 그린코퍼
휴대폰액세서리 356	<input type="checkbox"/> 라셀르	<input type="checkbox"/> 가쯔

Pivot Facet (Decision Tree)

Solr는 단일 Facet뿐만 아니라 아래와 같이 계층구조의 데이터의 Facet 기능도 제공합니다.

- Pivot Facet의 예

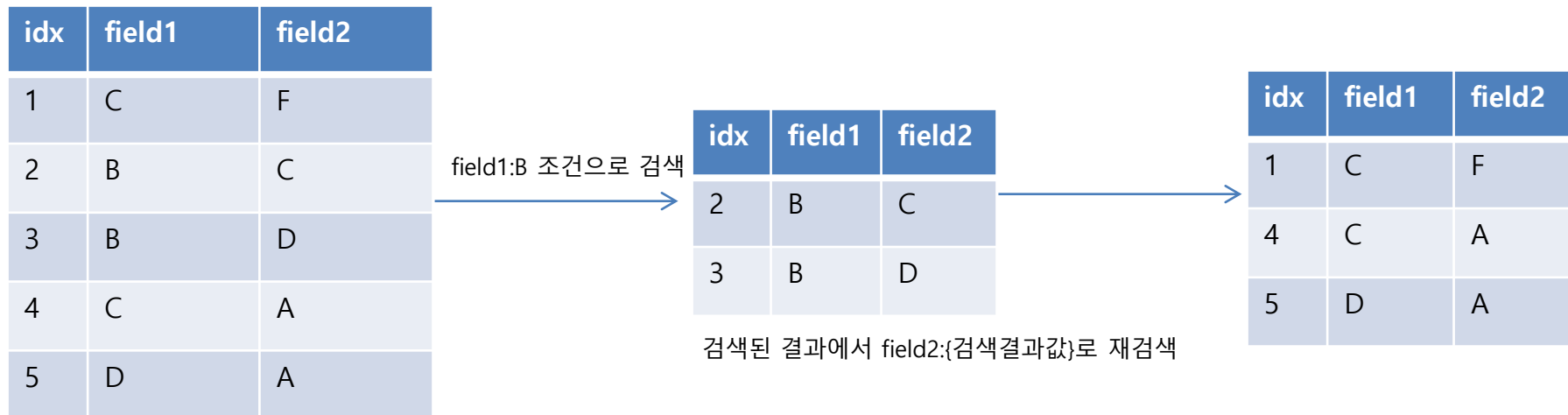
한번의 검색으로 3 Depth의 자료에 대한 개수를
Facet하여 다양한 서비스를 구축 가능합니다.



JOIN 검색 기능

검색된 결과를 가지고 다시 검색 조건으로 넣어 검색하는 기능입니다.
(결과내 재검색 기능하고는 다른 기능)

- JOIN 기능의 예



이용 사례

- SNS의 서비스의 경우 사용자의 1촌에서 2촌을 실시간으로 계산
- 데이터의 연관 관계를 Tree를 타고 내려가듯 추적하는 시스템의 경우 사용

Solr 검색엔진은 사용자와 아이템간의 좌표 정보를 이용하여 좌표 검색이 가능합니다.

좌표 검색은

점(Point) 대 점 (Point)

점(Point) 대 면 (Polygon)

면(Polygon) 대 면 (Polygon) 검색이 가능합니다.

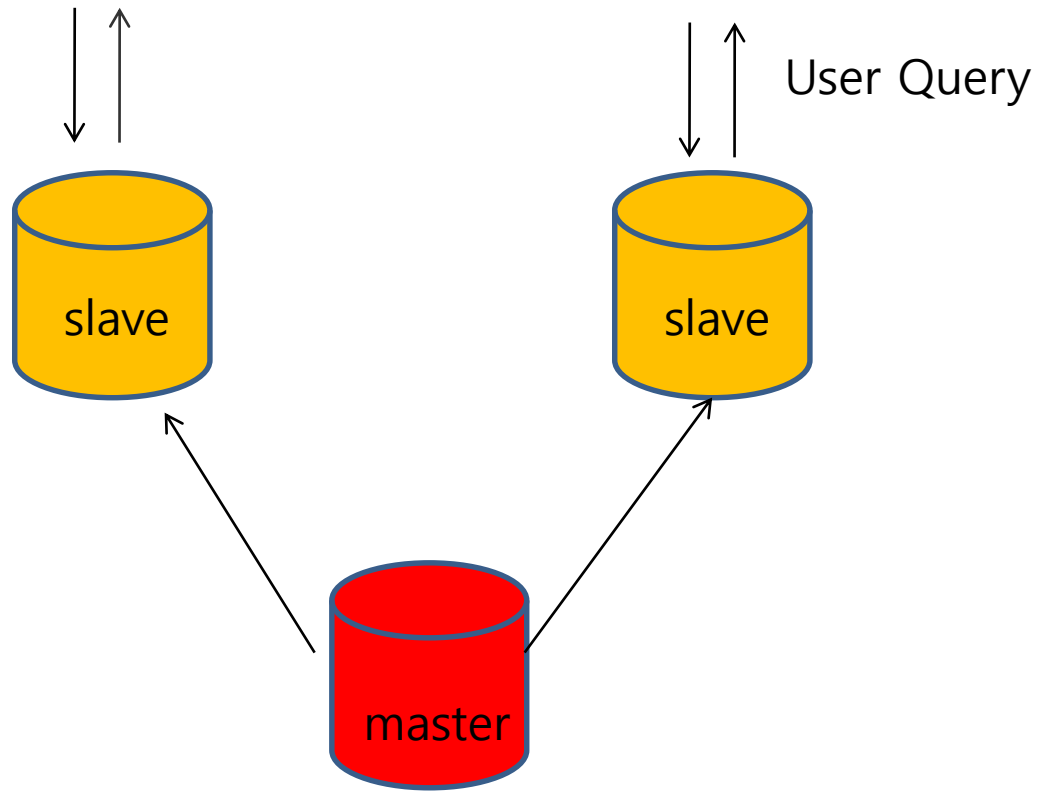




08 스케일 아웃

복제

복제(replication)



Replication 설정 방법

solrconfig에 설정

- Master

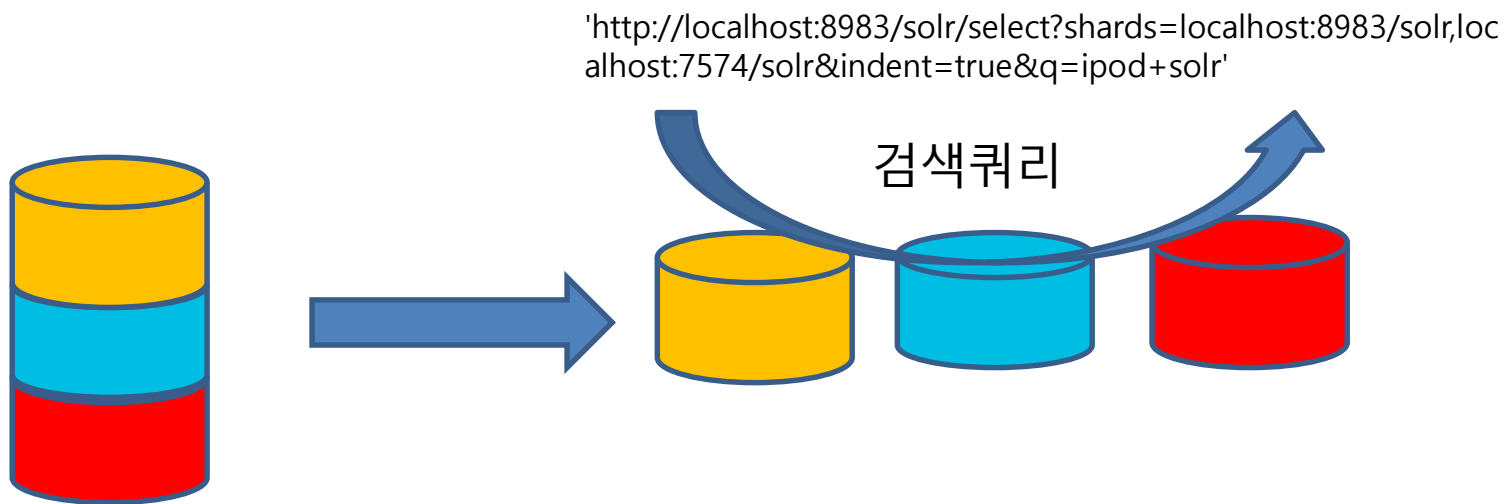
```
<requestHandler name="/replication" class="solr.ReplicationHandler" >  
  <lst name="master">  
    <str name="replicateAfter">commit</str>  
    <str name="replicateAfter">startup</str>  
    <str name="confFiles">schema.xml,stopwords.txt,synonyms.txt</str>  
  </lst>  
</requestHandler>
```

-Slave

```
<requestHandler name="/replication" class="solr.ReplicationHandler" >  
  <lst name="slave">  
    <str name="masterUrl">{마스터URL}/Shop/replication</str>  
    <str name="pollInterval">00:00:60</str>  
  </lst>  
</requestHandler>
```

샤딩(Sharding)

데이터를 횡으로 나누어 배치하는 전략



스케일 아웃 전략

