the search is over

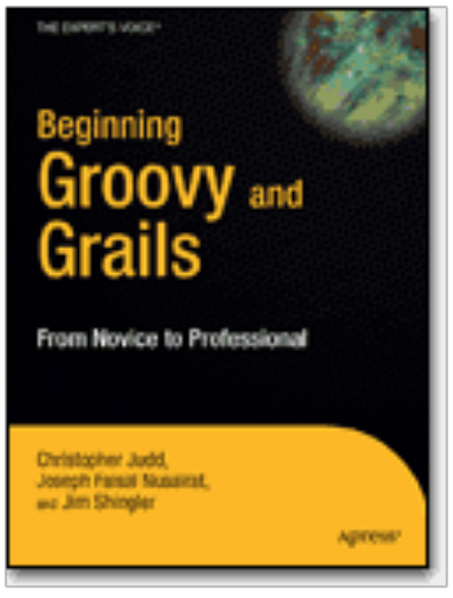
Christopher M. Judd

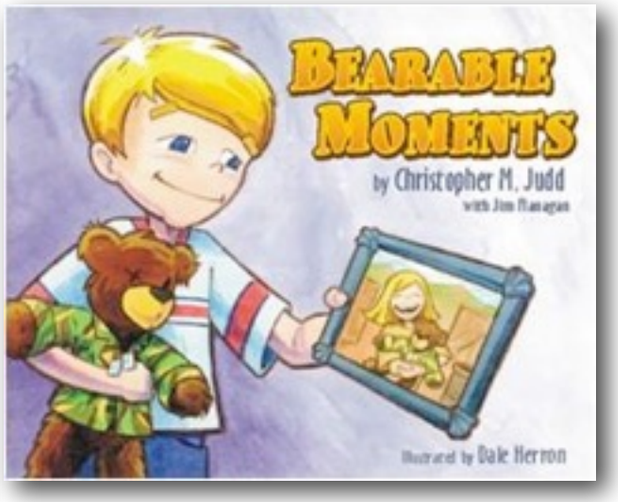
Christopher M. Judd

CTO and Partner at Manifest Solutions

Central Ohio Java Users Group leader

Columbus iPhone Developer User Group (CIDUG)





Customer Name	Product Name	Access	Action
abcde	alpha	Administrator	
efghi	gamma	Maintainer	
jlkhkjh	beta	Viewer	
jhfhjnm	beta	Administrator	
Netkj	beta	Maintainer	
kjhkl	beta	Viewer	
szvxcbv	beta	Administrator	
xcvnb	beta	Maintainer	
dfxgc	beta	Maintainer	
gdfch	beta	Maintainer	
dfghn	beta	Maintainer	
xvcbnv	beta	Maintainer	

Page 1 of 1

Displaying 1 - 12 of 12

solr - Google Search

https://www.google.com/search?q=geospatial&oq=geospatial&aqs=chrome..69i57j0l5.655j0j7&sourceid...

Google

solr

solr Remove
solrock
solrepublic
solr tutorial

+Christ... Share

About 4,970,000 results (0.28 seconds)

LucidWorks™ Search 2.7 - LucidWorks.com
Ad www.lucidworks.com/Search
Industry Leading Offering of Lucene **Solr** For Enterprise. Learn More!
LucidWorks has 141 followers on Google+

Product Overview — LucidWorks University
Download Center — LucidWorks Consulting

Solr Faceted Search - searchblox.com
Ad www.searchblox.com/
Setup and Search in minutes. Easy setup. Web based console.
Download - Pricing - Solutions

Apache Lucene - Apache Solr
lucene.apache.org/solr/ Lucene
SolrTM is the popular, blazing fast open source enterprise search platform from the Apache LuceneTM project. Its major features include powerful full-text search ...
You've visited this page many times. Last visit: 6/22/14

Tutorial
A copy of the tutorial for each version of Solr is included in the ...

Apache Lucene
SolrTM Documentation. New users are encouraged to start by ...

Download
SolrTM Downloads. Official releases are usually created ...

Solr Wiki
SolrInstall - Apache Lucene - SolrQuerySyntax - UpdateJSON

Features
Solr is a standalone enterprise search server with a REST-like ...

Solr Tutorial
Please run the browser showing this tutorial and the Solr server ...

More results from apache.org »

HowToContribute - Solr Wiki - General Wiki
wiki.apache.org/solr/HowToContribute Apache Software Foundation
May 18, 2014 - Please keep discussions about **Solr** on list so that everyone benefits. Emailing individual committers with questions about specific **Solr** issues is ...

Start

Search

Everywhere

Visual Studio 2013

Visual Studio 2013

Visual Studio Tools

Choose whether to use larger touch feedback

Show visual feedback when you touch the screen

Optimize visual display

Replace sounds with visual cues

Visual

visual studio

visual studio 2013 update 2

visual studio 2013

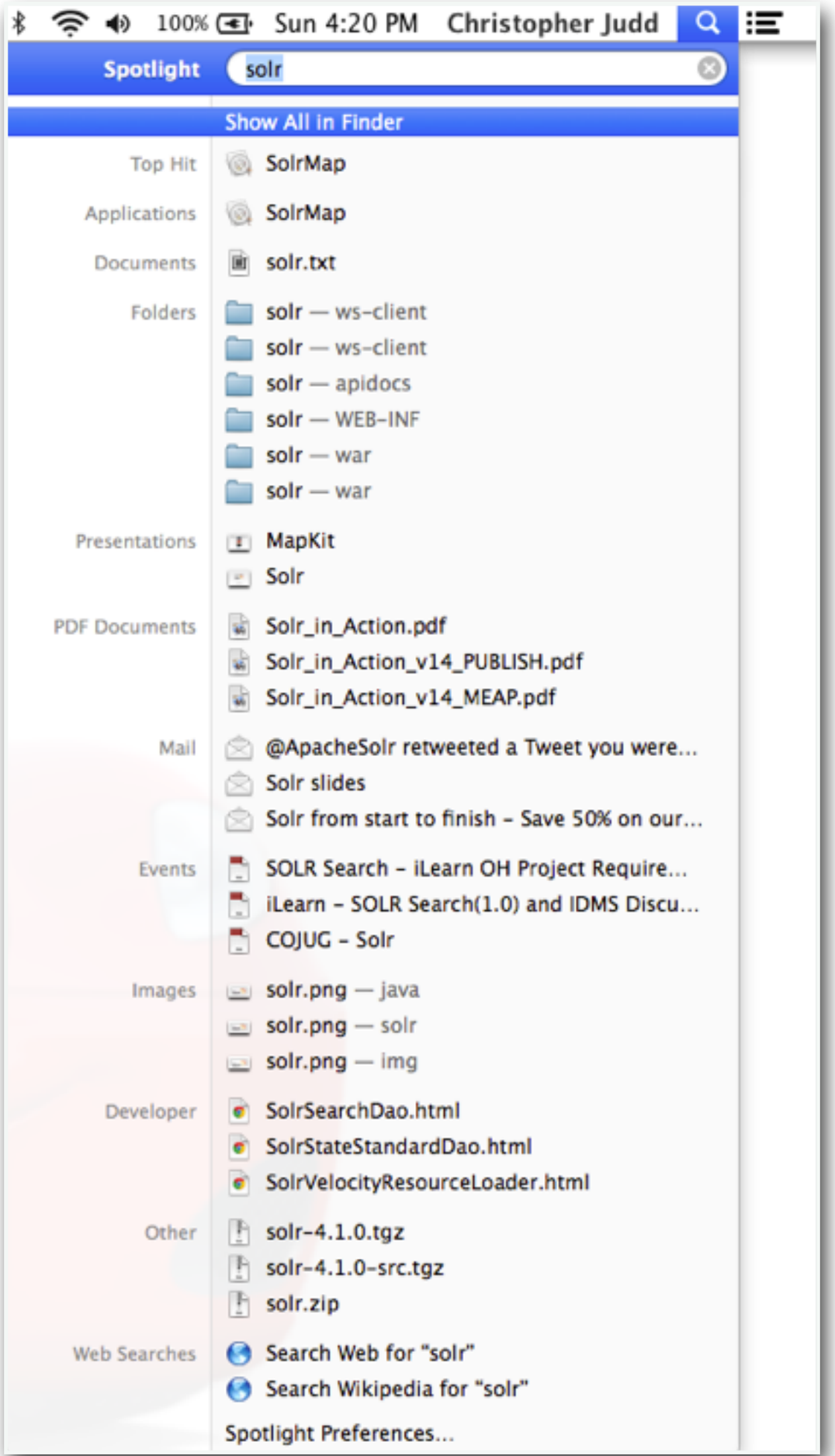
visual studio online

visual studio express

visual boy advance

visual basic

visual comfort


100% Sun 4:20 PM Christopher Judd
Spotlight
solr
Show All in Finder
Top Hit
SolrMap
Applications
SolrMap
Documents
solr.txt
Folders
solr — ws-client
solr — ws-client
solr — apidocs
solr — WEB-INF
solr — war
solr — war
Presentations
MapKit
Solr
PDF Documents
Solr_in_Action.pdf
Solr_in_Action_v14_PUBLISH.pdf
Solr_in_Action_v14_MEAP.pdf
Mail
@ApacheSolr retweeted a Tweet you were...
Solr slides
Solr from start to finish - Save 50% on our...
Events
SOLR Search - iLearn OH Project Require...
iLearn - SOLR Search(1.0) and IDMS Discu...
COJUG - Solr
Images
solr.png — java
solr.png — solr
solr.png — img
Developer
SolrSearchDao.html
SolrStateStandardDao.html
SolrVelocityResourceLoader.html
Other
solr-4.1.0.tgz
solr-4.1.0-src.tgz
solr.zip
Web Searches
Search Web for "solr"
Search Wikipedia for "solr"
Spotlight Preferences...

Java – ADT – /Users/cjudd/devl/tmp/android

java

Previous Choices	Classpath Variables – **Java**/Build Path
Editors	Constants.**java** – MainActivity/src/com/aad/hello/Constants.java
	MainActivity.**java** – MainActivity/src/com/aad/hello/MainActivity.java
Views	**Javadoc**
Perspectives	**Java**
	Java Browsing
	Java Type Hierarchy
Commands	**Java** Project... – New Java Project
	Java...
	Add **Java** Exception Breakpoint – Add a Java exception breakpoint
	Class... – New **Java** Class
	Close Perspective (**Java** Browsing) – Close the current perspective
Menus	**Java**...
	Add **Java** Exception Breakpoint...
	Generate **Java**doc...
	Open Attached **Java**doc – Open Attached Javadoc in a Browser
New	**Java** Project – Create a Java project
	Java Project from Existing Ant Buildfile – Create a Java project from an existing Ant buildfile on the file system
	Java Working Set – Creates a Java working set
	Class – Create a **Java** class
	Interface – Create a **Java** interface
Preferences	**Java**
	Javadoc – Java/Compiler
	Advanced – **Java**/Editor/Content Assist
	Appearance – **Java**
	Build Path – **Java**

Press '⌘3' to show all matches

Java Debug

N : String
DING : String
THORIZATION : Stri
THORIZATION_VAL
_URL : String
String
RL : String
D : String
String
L : String
FERENCE_NAME : S
ASK_BROKER_INSTA
CHECK_BROKER : St

StringExtensions.java
TokenCacheItem.java
UserInfo.java
WebRequestHandler.java
gen [Generated Java Files]
assets
bin
libs
res
target
AndroidManifest.xml
build.xml

Saved Filters
All messages (no filters)

Search for messages. Accepts Java regexes. Prefix with pid:, app:, tag: or text: to limit scope.

verbose

Le	Time	PID	TID	Application	Tag	Text

com.microsoft.adal – com.microsoft.adal/src

128M of 244M


AT&T
7:39 PM
ang
Cancel
TOP HITS
Angry Birds
APPLICATIONS
Angry Birds
Angry Birds
Angry Birds
Angry Birds
Angry Birds
Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M
.?123
space
Search

searching is easy

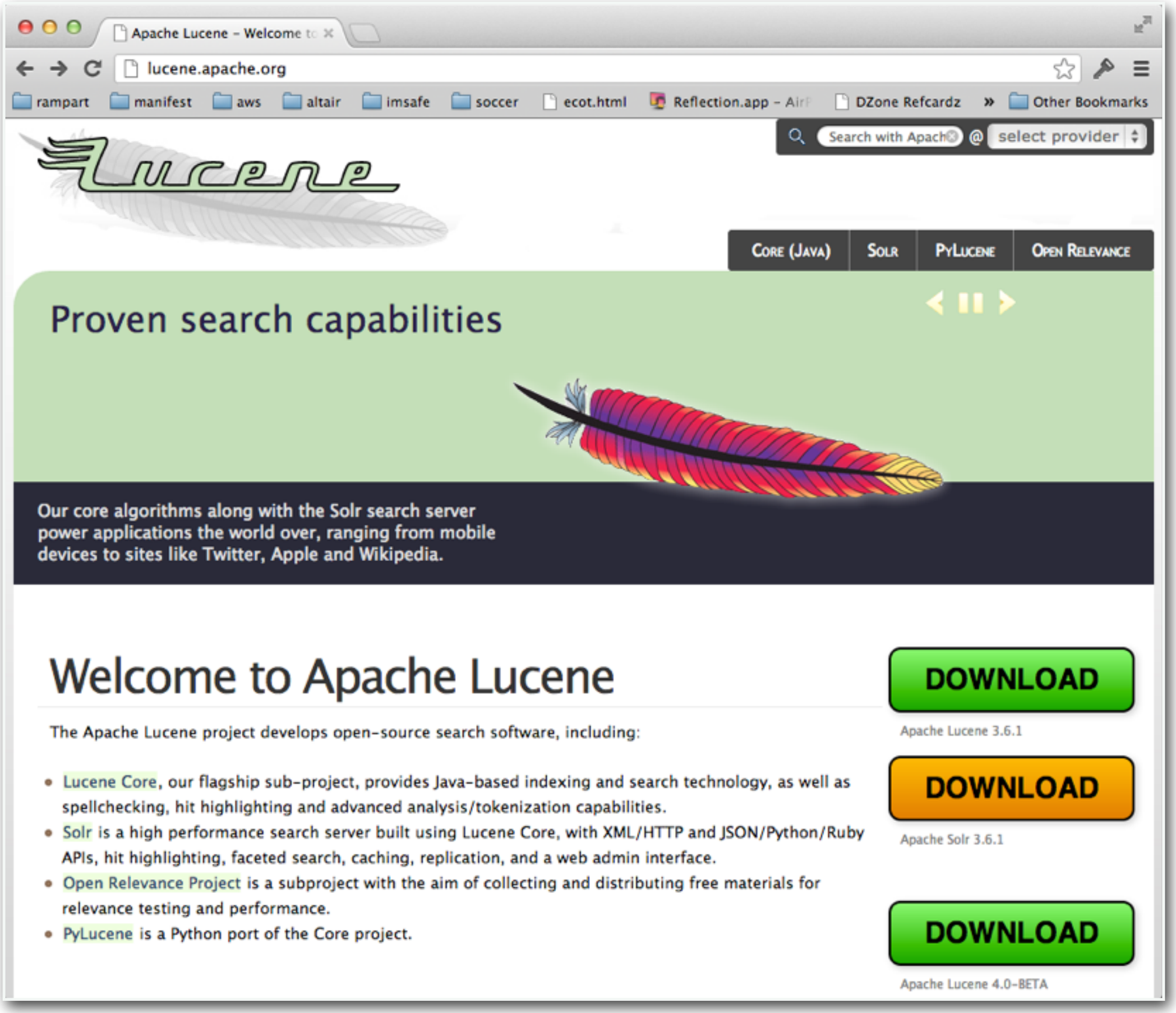
right?

```
select *
from products
where name = 'iPhone 5'
```

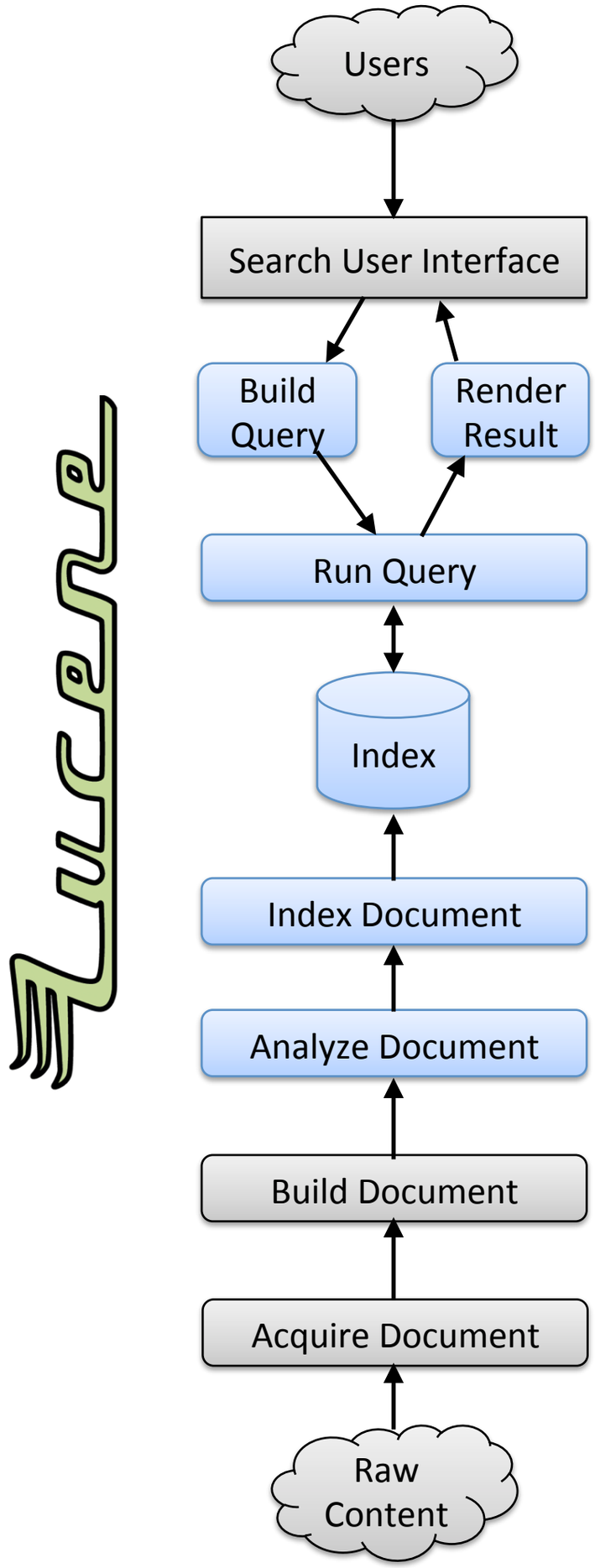
```
select *
from products
where description like '%iphone%'
```

```
select *
from products
where match(name, description)
      against('+iphone -case'
      in boolean mode);
```

Lucerne



http://lucene.apache.org/


Users
Search User Interface
Build Query
Render Result
Run Query
Index
Index Document
Analyze Document
Build Document
Acquire Document
Raw Content

```
public class InMemoryExample {

    public static void main(String[] args) throws CorruptIndexException, LockObtainFailedException, IOException, ParseException {
        // in-memory representation of the index
        RAMDirectory idx = new RAMDirectory();

        // Make an writer to create the index
        IndexWriterConfig config = new IndexWriterConfig(LUCENE_36, new StandardAnalyzer(LUCENE_36));
        IndexWriter writer = new IndexWriter(idx, config);

        // Add some Document objects containing quotes
        writer.addDocument(createDocument(
                  "Theodore Roosevelt", "It behooves every man to remember that the work of the "
                  + "critic, is of altogether secondary importance, and that, "
                  + "in the end, progress is accomplished by the man who does "
                  + "things."));
        writer.addDocument(createDocument(
                  "Friedrich Hayek", "The case for individual freedom rests largely on the "
                  + "recognition of the inevitable and universal ignorance "
                  + "of all of us concerning a great many of the factors on "
                  + "which the achievements of our ends and welfare depend."));
        writer.addDocument(createDocument(
                  "Ayn Rand", "There is nothing to take a man's freedom away from "
                  + "him, save other men. To be free, a man must be free "
                  + "of his brothers."));
        writer.addDocument(createDocument(
                  "Mohandas Gandhi", "" +
                  "Freedom is not worth having if it does not connote "
                  + "freedom to err."));

        writer.close();

        // Build an IndexSearcher using the in-memory index
        IndexReader reader = IndexReader.open(idx);
        IndexSearcher searcher = new IndexSearcher(reader);

        // Run some queries
        search(searcher, "freedom");
        search(searcher, "free");
        search(searcher, "progress or achievements");

        searcher.close();
    }
```

```
        search(searcher, "progress or achievements");

        searcher.close();
    }

    /**
     * Make a Document object with an un-indexed title field and an indexed content field.
     */
    private static Document createDocument(String title, String content) {
        Document doc = new Document();

        // Add the title as an unindexed field...
        doc.add(new Field("title", title, Field.Store.YES, Field.Index.NOT_ANALYZED));
        doc.add(new Field("content", content, Field.Store.YES, Field.Index.ANALYZED));

        return doc;
    }

    /**
     * Searches for the given string in the "content" field
     */
    private static void search(IndexSearcher searcher, String queryString) throws ParseException, IOException {

        // Build a Query object
        Query query = new QueryParser(LUCENE_36, "content", new StandardAnalyzer(LUCENE_36)).parse(queryString);

        TopDocsCollector collector = TopScoreDocCollector.create(10, true);
        searcher.search(query, collector);

        if (collector.getTotalHits() == 0) {
            System.out.println("No matches were found for \"" + queryString + "\"");
        } else {
            System.out.println("Hits for \"" + queryString + "\" were found in quotes by:");

            ScoreDoc[] hits = collector.topDocs().scoreDocs;
            for (ScoreDoc hit : hits) {
                Document doc = searcher.doc(hit.doc);
                System.out.println(" - " + doc.get("title"));
            }

        }
        System.out.println();
    }

}
```

```
        search(searcher, "progress or achievements");

        searcher.close();
    }

    /**
     * Make a Document object with an un-indexed title field and an indexed content field.
     */
    private static Document createDocument(String title, String content) {
        Document doc = new Document();

        // Add the title as an unindexed field...
        doc.add(new Field("title", title, Field.Store.YES, Field.Index.NOT_ANALYZED));
        doc.add(new Field("content", content, Field.Store.YES, Field.Index.ANALYZED));

        return doc;
    }

    /**
     * Searches for the given string in the "content" field
     */
    private static void search(IndexSearcher searcher, String queryString) throws ParseException, IOException {

        // Build a Query object
        Query query = new QueryParser(LUCENE_36, "content", new StandardAnalyzer(LUCENE_36)).parse(queryString);

        TopDocsCollector collector = TopScoreDocCollector.create(10, true);
        searcher.search(query, collector);

        if (collector.getTotalHits() == 0) {
            System.out.println("No matches
        } else {
            System.out.println("Hits for \"

            ScoreDoc[] hits = collector.top
            for (ScoreDoc hit : hits) {
                Document doc = searcher.do
                System.out.println(" - " +
            }

        }
        System.out.println();
    }

}
```

```
Hits for "freedom" were found in quotes by:
 - Mohandas Gandhi
 - Ayn Rand
 - Friedrich Hayek

Hits for "free" were found in quotes by:
 - Ayn Rand

Hits for "progress or achievements" were found in quotes by:
 - Theodore Roosevelt
 - Friedrich Hayek
```

Index

Document

Field
title

Value

Term Theodore

Term Roosevelt

Field
content

Value

Term It

Term behooves

Term every

. . .

Document

Field
title

Value

Term Fredrich

Term Hayek

Field
content

Value

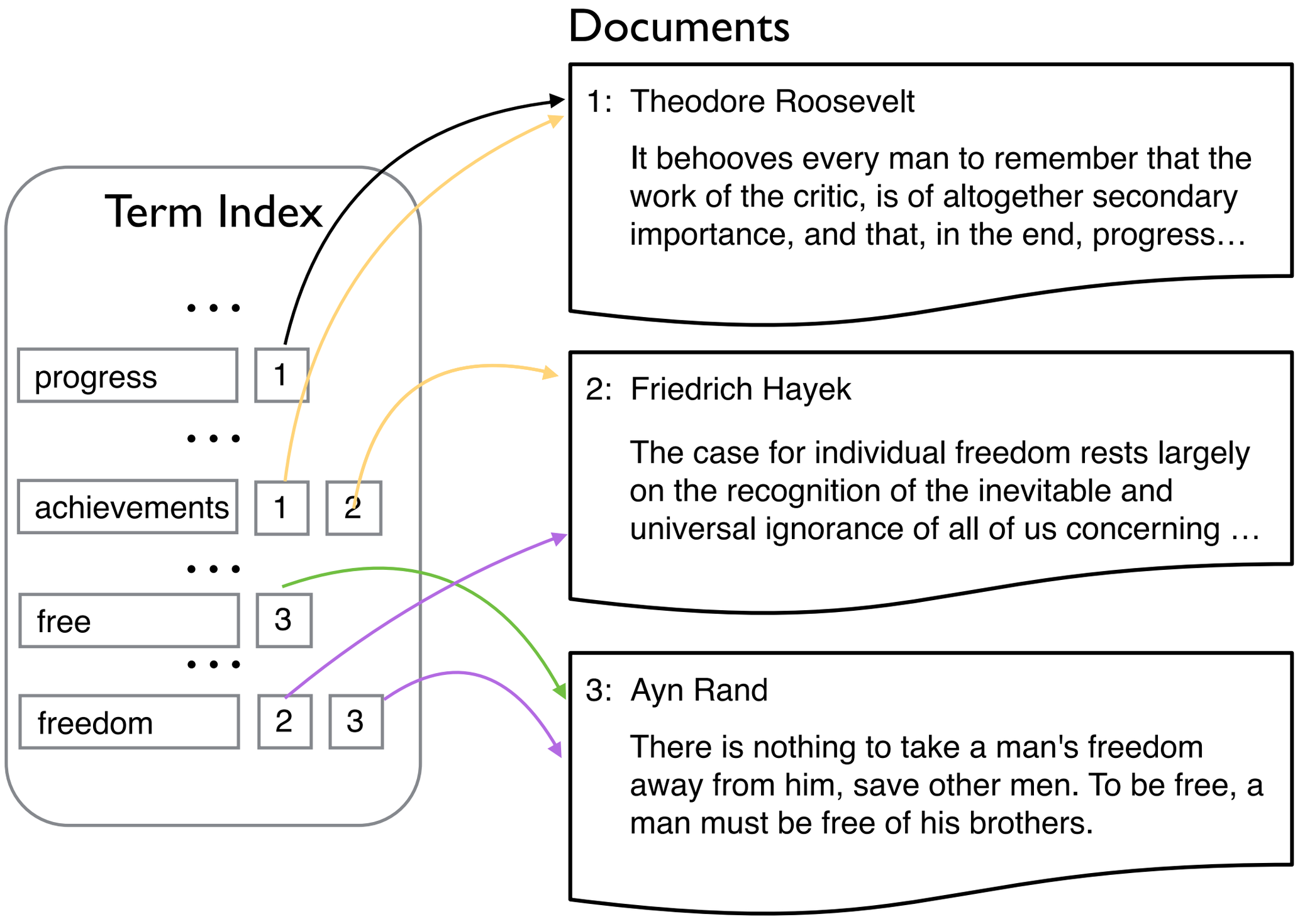
Term The

Term case

Term for

. . .

. . .


Documents
Term Index
progress
1
achievements
1
2
free
3
freedom
2
3
1: Theodore Roosevelt
It behooves every man to remember that the work of the critic, is of altogether secondary importance, and that, in the end, progress…
2: Friedrich Hayek
The case for individual freedom rests largely on the recognition of the inevitable and universal ignorance of all of us concerning …
3: Ayn Rand
There is nothing to take a man's freedom away from him, save other men. To be free, a man must be free of his brothers.

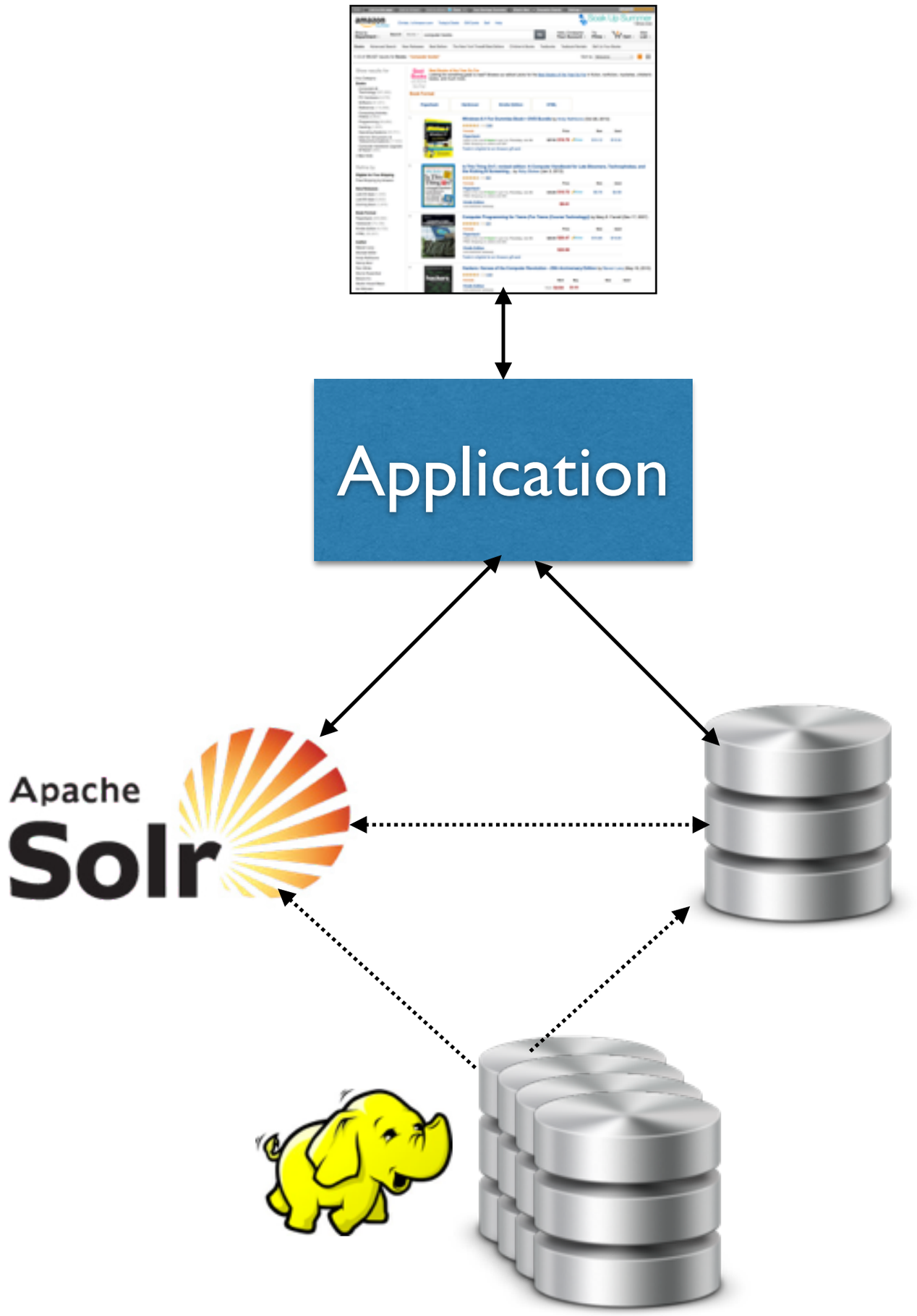
- popular
- powerful

- Java only
- single threaded writes
- lots of boiler plate code
- difficult to debug and test indexes
- no user interface

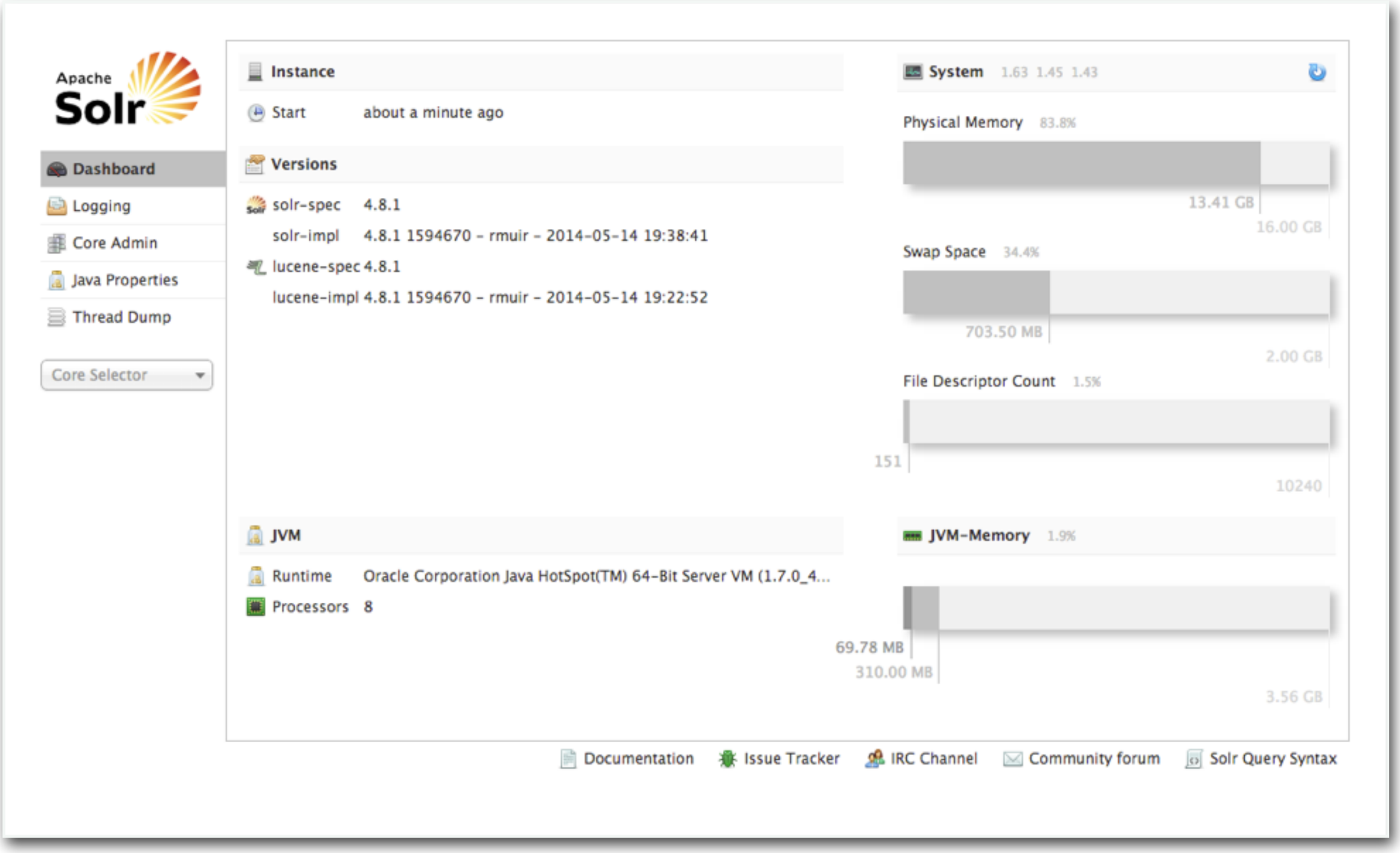

Apache
Solr

http://lucene.apache.org/solr/

- scalable
- caching
- highlighting
- boosting
- facets
- clustering
- database integration
- rich document (Word, PDF) indexing
- geospatial search
- REST-like HTTP/XML and JSON API


Application
Apache
Solr
datasources

java -jar start.jar



http://localhost:8983/solr/

Lab 1

1. Install Solr >= 4.9
2. Start Solr
3. Browse to http://localhost:8983/solr/

Basic Querying

Apache Solr

- Dashboard
- Logging
- Core Admin
- Java Properties
- Thread Dump

collection1

- Overview
- Analysis
- Dataimport
- Documents
- Files
- Ping
- Plugins / Stats
- Query
- Replication
- Schema Browser

Request-Handler (qt)

/select

— common —

q

:

fq

sort

start, rows

0 10

fl

df

Raw Query Parameters

key1=val1&key2=val2

wt

json

☑ indent

☐ debugQuery

☐ dismax

☐ edismax

☐ hl

☐ facet

☐ spatial

☐ spellcheck

Execute Query

http://localhost:8983/solr/collection1/select?q=*%3A*&wt=json&indent=true


```
{
  "responseHeader": {
    "status": 0,
    "QTime": 1,
    "params": {
      "indent": "true",
      "q": "*:*",
      "_": "1403522465107",
      "wt": "json"
    }
  },
  "response": {
    "numFound": 0,
    "start": 0,
    "docs": []
  }
}
```


Documentation Issue Tracker IRC Channel Community forum Solr Query Syntax

http://localhost:8983/solr/collection1/select?q=*%3A*&wt=json&indent=true


```
{
  "responseHeader": {
    "status": 0,
    "QTime": 1,
    "params": {
      "indent": "true",
      "q": "*:*",
      "_": "1403522465107",
      "wt": "json"
```

request handler

output format

http://localhost:8983/solr/collection1/select?q=*:*&wt=xml

index

query for everything

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
<lst name="responseHeader"><int name="status">0</int><int
name="QTime">2</int><lst name="params"><str name="wt">xml</str><str
name="q">*:*</str></lst></lst><result name="response" numFound="0"
start="0"></result>
</response>
```

/select?q=*:*&wt=xml&indent=on

pretty print

```
<?xml version="1.0" encoding="UTF-8"?>
<response>

<lst name="responseHeader">
  <int name="status">0</int>
  <int name="QTime">1</int>
  <lst name="params">
    <str name="indent">on</str>
    <str name="wt">xml</str>
    <str name="q">*:*</str>
  </lst>
</lst>
<result name="response" numFound="0" start="0">
</result>
</response>
```

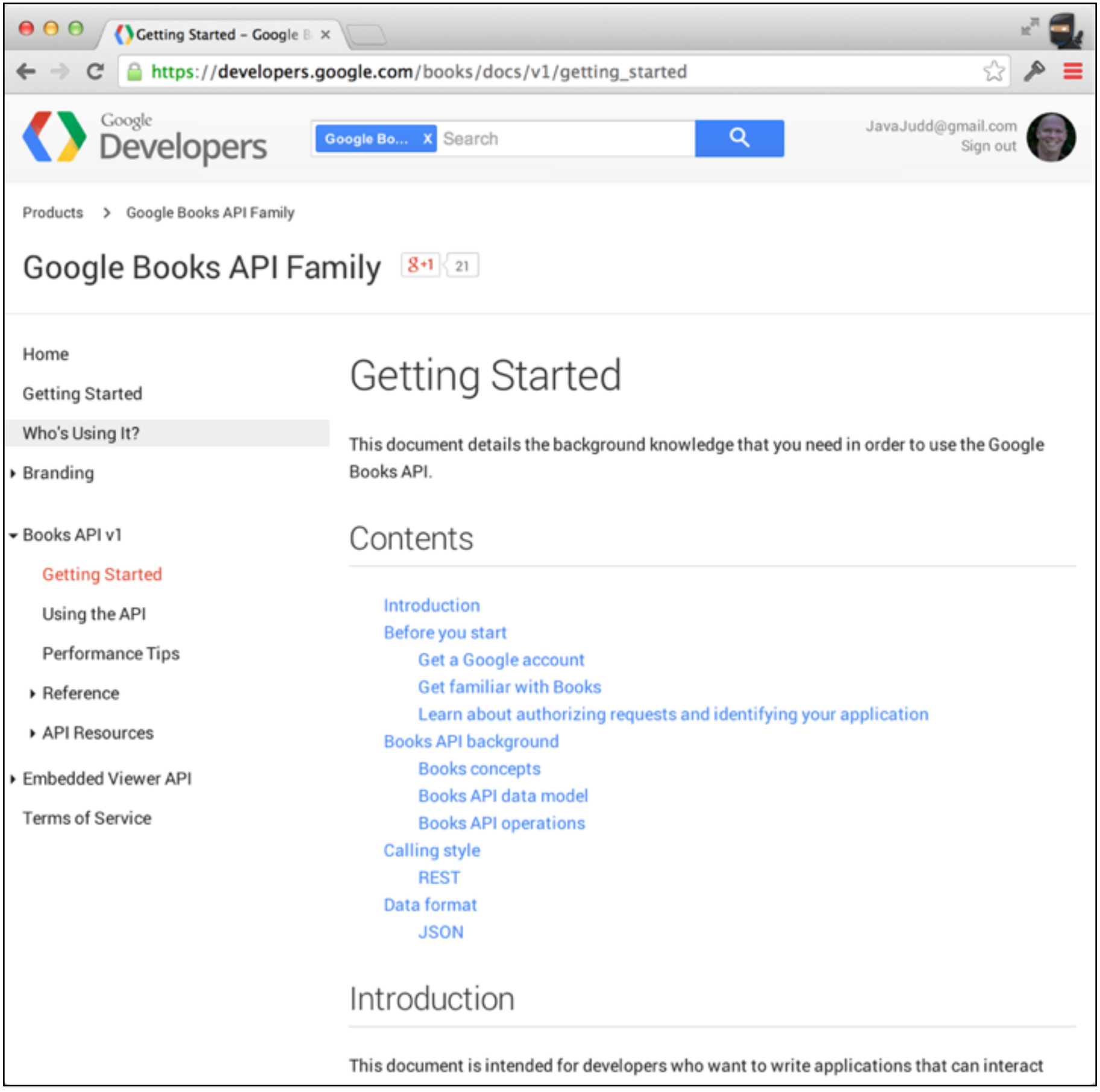
```
/select?q=*:*&wt=json&indent=on
```

json


```
{
  "responseHeader":{
    "status":0,
    "QTime":0,
    "params":{
      "indent":"on",
      "wt":"json",
      "q":"*:*"}},
"response":{"numFound":0,"start":0,"docs":[]
}}
```

Index Data

- XML
- JSON
- CSV
- Database



https://developers.google.com/books/docs/v1/getting_started

```
<add>
  <doc>
    <field name="id">ka2VUBqHiWkC</field>
    <field name="title">Effective Java</field>
    <field name="author">Joshua Bloch</field>
    <field name="author_txt">Joshua Bloch</field>
    <field name="pages_i">368</field>
    <field name="saleable_b">true</field>
    <field name="description">Are you looking for a deeper understanding of Java …</field>
    <field name="price">25.49</field>
  </doc>
  <doc>
    <field name="id">-SYM4PW-YAgC</field>
    <field name="title">The Religion of Java</field>
    <field name="author">Clifford Geertz</field>
    <field name="author_txt">Clifford Geertz</field>
    <field name="pages_i">392</field>
    <field name="saleable_b">false</field>
    <field name="description">Written with a rare combination of analysis and  …</field>
  </doc>
</add>
```

```
[
    {
        "id": "ka2VUBqHiWkC",
        "title": "Effective Java",
        "author": "Joshua Bloch",
        "author_txt": [
            "Joshua Bloch"
        ],
        "page_i": 368,
        "saleable_b": true,
        "description": "Are you looking for a deeper understanding of the Java …”,
        "price": 25.49
    },
    {
        "id": "-SYM4PW-YAgC",
        "title": "The Religion of Java",
        "author": "Clifford Geertz",
        "author_txt": [
            "Clifford Geertz"
        ],
        "page_i": 392,
        "saleable_b": false,
        "description": "Written with a rare combination of analysis and speculation …”
    }
]
```

```
curl /update/json --data-binary <file> -H 'Content-type:application/json'
```


```
{"responseHeader":{"status":0,"QTime":22}}
```


```
curl /update/json?softCommit=true
```


```
<?xml version="1.0" encoding="UTF-8"?>
<response>
<lst name="responseHeader"><int name="status">0</int><int name="QTime">28</int></lst>
</response>
```

Lab 2

1. Load Google book data
2. Perform a search to determine how many books
3. Return a result in XML
4. Return a human readable result in XML
5. Return a result in JSON
6. Return a human readable result in JSON

More Querying

/select?q=*:*&wt=json&indent=true


```
{
  "responseHeader": {
    "status": 0,
    "QTime": 0,
    "params": {
      "indent": "true",
      "q": "*:*",
      "_": "1403710200465",
      "wt": "json"
    }
  },
  "response": {
    "numFound": 1,
    "start": 0,
    "docs": [
      {
        "id": "ka2VUBqHiWkC",
        "title": [
          "Effective Java"
        ],
        "author": "Joshua Bloch",
        "author_s": "Joshua Bloch",
        "author_txt": [
          "Joshua Bloch"
        ],
        "saleable_b": true,
        "description": "Are you looking for a deeper understanding of the Java ...",
       "_version_": 1471896715487346700
      }
    ]
  }
}
```

Limit Fields

/select?q=*:*&**fl=name id**&wt=json&indent=true


```
{
  "responseHeader":{
    "status":0,
    "QTime":0,
    "params":{
      "fl":"name id",
      "indent":"true",
      "q":"*:*",
      "wt":"json"}},
  "response":{"numFound":10,"start":0,"docs":[
      { "id":"ka2VUBqHiWkC", "name":"Effective Java"},
      { "id":"-SYM4PW-YAgC", "name":"The Religion of Java"},
      { "id":"mB_92VqJbsMC", "name":"Java Threads"},
      { "id":"Ql6QgWf6i7cC", "name":"Thinking in Java"},
      { "id":"zuGy-V3Nk4AC", "name":"Java SE 7 Programming Essentials"},
      { "id":"mvzgNSmHEUAC", "name":"Java in a Nutshell"},
      { "id":"gJEC2q7DzpQC", "name":"The History of Java"},
      { "id":"vvg7fN_HScAC", "name":"Advanced Java Networking"},
      { "id":"pnwTLvCJKh0C", "name":"Java Programming"},
      { "id":"Y0lDBsh7J9kC", "name":"Learning Java"}]
  }}
```

Limit Results

/select?q=*:*&**rows=5**&fl=name+id&wt=json&indent=true


```
{
  "responseHeader":{
    "status":0,
    "QTime":0,
    "params":{
      "fl":"name id",
      "indent":"true",
      "q":"*:*",
      "wt":"json",
      "rows":"5"}},
  "response":{"numFound":10,"start":0,"docs":[
      { "id":"ka2VUBqHiWkC", "name":"Effective Java"},
      { "id":"-SYM4PW-YAgC", "name":"The Religion of Java"},
      { "id":"mB_92VqJbsMC", "name":"Java Threads"},
      { "id":"Ql6QgWf6i7cC", "name":"Thinking in Java"},
      { "id":"zuGy-V3Nk4AC", "name":"Java SE 7 Programming Essentials"}]
  }}
```

Pagination

/select?q=*:*&**start=5&rows=5**&fl=name+id&wt=json&indent=true


```
{
  "responseHeader":{
    "status":0,
    "QTime":0,
    "params":{
      "fl":"name id",
      "indent":"true",
      "start":"5",
      "q":"*:*",
      "wt":"json",
      "rows":"5"}},
  "response":{"numFound":10,"start":5,"docs":[
      { "id":"mvzgNSmHEUAC", "name":"Java in a Nutshell"},
      { "id":"gJEC2q7DzpQC", "name":"The History of Java"},
      { "id":"vvg7fN_HScAC", "name":"Advanced Java Networking"},
      { "id":"pnwTLvCJKh0C", "name":"Java Programming"},
      { "id":"Y0lDBsh7J9kC", "name":"Learning Java"}]
  }}
```

Query Field

/select?**q=name:effective**&wt=json&indent=true


```
{
  "responseHeader":{
    "status":0,
    "QTime":1,
    "params":{
      "fl":"name id",
      "indent":"true",
      "q":"name:effective",
      "wt":"json"}},
"response":{"numFound":1,"start":0,"docs":[
    { "id":"ka2VUBqHiWkC", "name":"Effective Java"}]
}}
```

Query Multiple Fields

/select?**q=name:java+AND+saleable_b:true**
&fl=name+id+saleable_b&wt=json&indent=true


```
{
  "responseHeader":{
    "status":0,
    "QTime":0,
    "params":{
      "fl":"name id saleable_b",
      "indent":"true",
      "q":"name:java AND saleable_b:true\n",
      "wt":"json"}},
"response":{"numFound":6,"start":0,"docs":[
    { "id":"ka2VUBqHiWkC", "name":"Effective Java", "saleable_b":true},
    { "id":"mB_92VqJbsMC", "name":"Java Threads", "saleable_b":true},
    { "id":"Y0lDBsh7J9kC", "name":"Learning Java", "saleable_b":true},
    { "id":"mvzgNSmHEUAC", "name":"Java in a Nutshell", "saleable_b":true},
    { "id":"gJEC2q7DzpQC", "name":"The History of Java", "saleable_b":true},
    { "id":"zuGy-V3Nk4AC", "name":"Java SE 7 Programming Essentials", "saleable_b":true}]
}}
```

Query Ranges

/select?**q=pages_i:[*+TO+400]**
&fl=name+id+pages_i&wt=json&indent=true

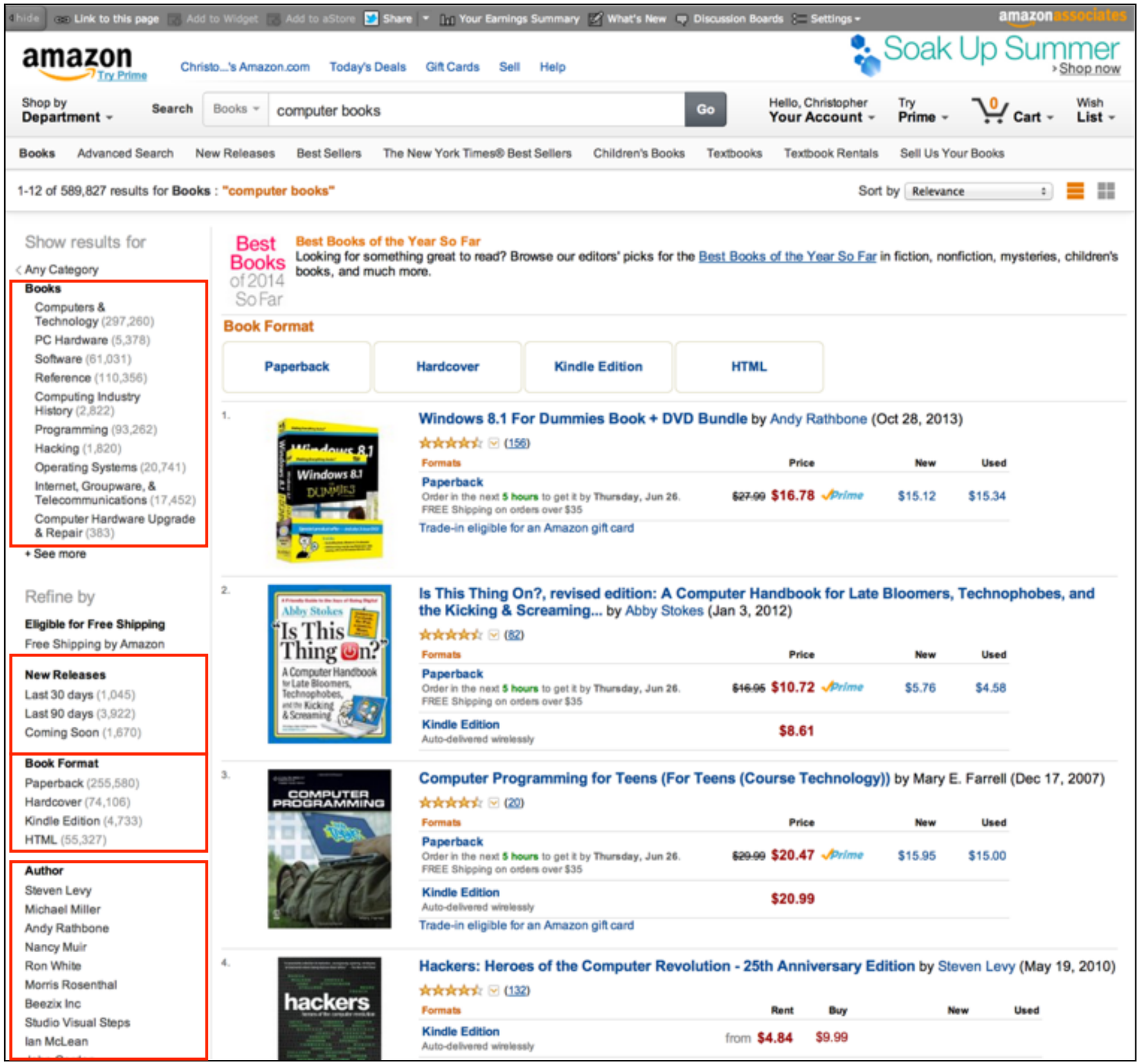

```
{
  "responseHeader":{
    "status":0,
    "QTime":0,
    "params":{
      "fl":"name id pages_i",
      "indent":"true",
      "q":"pages_i:[* TO 400]",
      "wt":"json"}},
  "response":{"numFound":5,"start":0,"docs":[
      { "id":"ka2VUBqHiWkC", "name":"Effective Java", "pages_i":368},
      { "id":"-SYM4PW-YAgC", "name":"The Religion of Java", "pages_i":392},
      { "id":"mB_92VqJbsMC", "name":"Java Threads", "pages_i":340},
      { "id":"zuGy-V3Nk4AC", "name":"Java SE 7 Programming Essentials", "pages_i":336},
      { "id":"vvg7fN_HScAC", "name":"Advanced Java Networking", "pages_i":399}]
  }}
```

Sorting

```
/select?q=pages_i:[*+TO+400]&sort=pages_i+asc
   &fl=name+id+pages_i&wt=json&indent=true
```


```
{
  "responseHeader":{
    "status":0,
    "QTime":0,
    "params":{
      "fl":"name id pages_i",
      "sort":"pages_i asc",
      "indent":"true",
      "q":"pages_i:[* TO 400]",
      "wt":"json"}},
  "response":{"numFound":5,"start":0,"docs":[
      { "id":"zuGy-V3Nk4AC", "name":"Java SE 7 Programming Essentials", "pages_i":336},
      { "id":"mB_92VqJbsMC", "name":"Java Threads", "pages_i":340},
      { "id":"ka2VUBqHiWkC", "name":"Effective Java", "pages_i":368},
      { "id":"-SYM4PW-YAgC", "name":"The Religion of Java", "pages_i":392},
      { "id":"vvg7fN_HScAC", "name":"Advanced Java Networking", "pages_i":399}]
  }}
```

Facets


hide
Link to this page
Add to Widget
Add to aStore
Share
Your Earnings Summary
What's New
Discussion Boards
Settings
amazonassociates
amazon
Try Prime
Christo...'s Amazon.com
Today's Deals
Gift Cards
Sell
Help
Soak Up Summer
Shop now
Shop by Department
Search
Books
computer books
Go
Hello, Christopher
Your Account
Try Prime
0 Cart
Wish List
Books
Advanced Search
New Releases
Best Sellers
The New York Times® Best Sellers
Children's Books
Textbooks
Textbook Rentals
Sell Us Your Books
1-12 of 589,827 results for Books : "computer books"
Sort by Relevance
Show results for
Any Category
Books
Computers & Technology (297,260)
PC Hardware (5,378)
Software (61,031)
Reference (110,356)
Computing Industry History (2,822)
Programming (93,262)
Hacking (1,820)
Operating Systems (20,741)
Internet, Groupware, & Telecommunications (17,452)
Computer Hardware Upgrade & Repair (383)
+ See more
Refine by
Eligible for Free Shipping
Free Shipping by Amazon
New Releases
Last 30 days (1,045)
Last 90 days (3,922)
Coming Soon (1,670)
Book Format
Paperback (255,580)
Hardcover (74,106)
Kindle Edition (4,733)
HTML (55,327)
Author
Steven Levy
Michael Miller
Andy Rathbone
Nancy Muir
Ron White
Morris Rosenthal
Beezix Inc
Studio Visual Steps
Ian McLean
Best Books of 2014 So Far
Best Books of the Year So Far
Looking for something great to read? Browse our editors' picks for the Best Books of the Year So Far in fiction, nonfiction, mysteries, children's books, and much more.
Book Format
Paperback
Hardcover
Kindle Edition
HTML
1.
Windows 8.1 For Dummies Book + DVD Bundle by Andy Rathbone (Oct 28, 2013)
(156)
Formats
Price
New
Used
Paperback
Order in the next 5 hours to get it by Thursday, Jun 26.
FREE Shipping on orders over $35
$27.99 $16.78 Prime
$15.12
$15.34
Trade-in eligible for an Amazon gift card
2.
Is This Thing On?, revised edition: A Computer Handbook for Late Bloomers, Technophobes, and the Kicking & Screaming... by Abby Stokes (Jan 3, 2012)
(82)
Formats
Price
New
Used
Paperback
Order in the next 5 hours to get it by Thursday, Jun 26.
FREE Shipping on orders over $35
$16.95 $10.72 Prime
$5.76
$4.58
Kindle Edition
Auto-delivered wirelessly
$8.61
3.
Computer Programming for Teens (For Teens (Course Technology)) by Mary E. Farrell (Dec 17, 2007)
(20)
Formats
Price
New
Used
Paperback
Order in the next 5 hours to get it by Thursday, Jun 26.
FREE Shipping on orders over $35
$29.99 $20.47 Prime
$15.95
$15.00
Kindle Edition
Auto-delivered wirelessly
$20.99
Trade-in eligible for an Amazon gift card
4.
Hackers: Heroes of the Computer Revolution - 25th Anniversary Edition by Steven Levy (May 19, 2010)
(132)
Formats
Rent
Buy
New
Used
Kindle Edition
Auto-delivered wirelessly
from $4.84
$9.99

Facets

/select?q=*:*&rows=0&wt=json&indent=true
&**facet=true**&**facet.field=cat**


```
{
    "facet_counts": {
        "facet_dates": {},
        "facet_fields": {
            "cat": [
                "Computers", 8,
                "Java (Indonesia)", 1,
                "Religion", 1
            ]
        },
        "facet_queries": {},
        "facet_ranges": {}
    },
    "response": {
        "docs": [], "numFound": 10, "start": 0
    },
    "responseHeader": {
        "QTime": 1,
        "params": {
            "facet": "true", "facet.field": "cat", "indent": "true", "q":
             "*:*", "rows": "0", "wt": "json"
        },
        "status": 0
    }
}
```

Highlighting

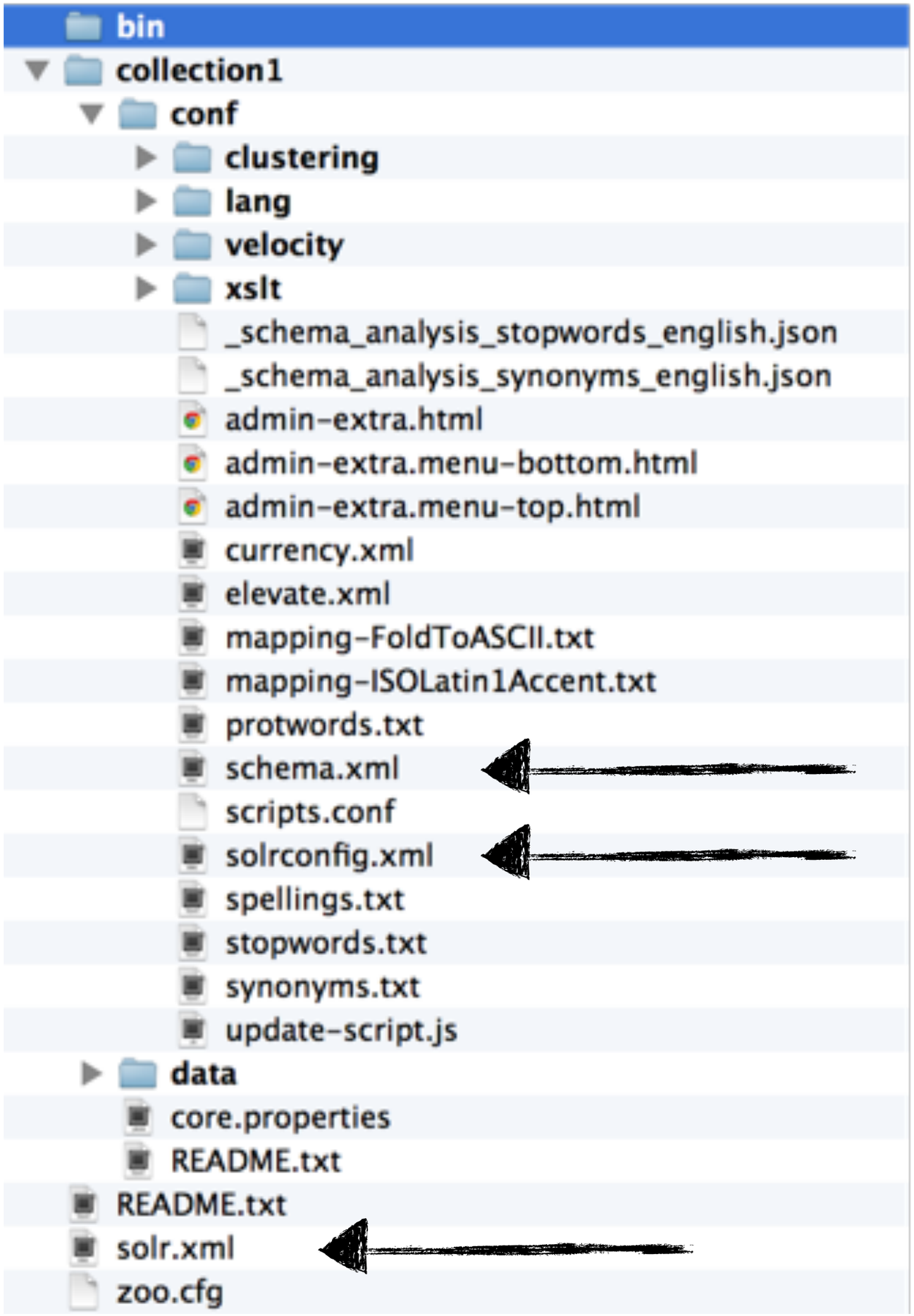
```
/select?q=description:java AND title:"Java in a Nutshell"&wt=json&indent=true
                                    &hl=true
                                  &hl.fl=description
                                &hl.simple.pre=<em>
                               &hl.simple.post=</em>
```


```
{
  "responseHeader":{
    "status":0,
    "QTime":28,
    "params":{
      "indent":"true",
      "q":"description:java AND title:\"Java in a Nutshell\" ",
      "hl.simple.pre":"<em>",
      "hl.simple.post":"<em>",
      "hl.fl":"description",
      "wt":"json",
      "hl":"true"}},
  "response":{"numFound":1,"start":0,"docs":[
      {
        "id":"mvzgNSmHEUAC",
        "title":["Java in a Nutshell"],
        "author":"David Flanagan",
        "author_s":"David Flanagan",
        "author_txt":["David Flanagan"],
        "page_i":1224,
        "saleable_b":true,
        "description":"Aimed for programmers, offers an introduction to Java 5.0 ...",
        "_version_":1471896715499929600}]
  },
  "highlighting":{
    "mvzgNSmHEUAC":{
      "description":["Aimed for programmers, offers an introduction to <em>Java</em> 5.0, covering topics such as
generics"]}}}
```

Lab 3

1. Search for all books but only return the id, name and author
2. Return a result with all books
3. Search for any books with java
4. Search for any books with out java
5. Search for any books with java in the title (name)
6. Search for any books with java in the title and are available to purchase
7. Search for books between 100 and 200 pages
8. Sort all the books based on the author's name
9. Determine how many books there in each category
10. Highlight the word java in the description of all the books

Configurations


bin
collection1
conf
clustering
lang
velocity
xslt
_schema_analysis_stopwords_english.json
_schema_analysis_synonyms_english.json
admin-extra.html
admin-extra.menu-bottom.html
admin-extra.menu-top.html
currency.xml
elevate.xml
mapping-FoldToASCII.txt
mapping-ISOLatin1Accent.txt
protwords.txt
schema.xml
scripts.conf
solrconfig.xml
spellings.txt
stopwords.txt
synonyms.txt
update-script.js
data
core.properties
README.txt
README.txt
solr.xml
zoo.cfg

schema.xml

```
<schema>

  <field name="id"   type="string"       indexed="true" stored="true"  multiValued="false" required="true" />
  <field name="name" type="text_general" indexed="true" stored="true"/>
  <field name="cat"  type="string"       indexed="true" stored="true"  multiValued="true"/>
  <field name="text" type="text_general" indexed="true" stored="false" multiValued="true"/>

  <dynamicField name="*_i"   type="int"          indexed="true"  stored="true"/>
  <dynamicField name="*_s"   type="string"       indexed="true"  stored="true" />
  <dynamicField name="*_ss"  type="string"       indexed="true"  stored="true"    multiValued="true"/>
  <dynamicField name="*_t"   type="text_general" indexed="true"  stored="true"/>
  <dynamicField name="*_txt" type="text_general" indexed="true"  stored="true"    multiValued="true"/>
  <dynamicField name="*_b"   type="boolean"      indexed="true"  stored="true"/>
  <dynamicField name="*_p"   type="location"     indexed="true"  stored="true"/>

  <uniqueKey>id</uniqueKey>

  <copyField source="cat"    dest="text"/>
  <copyField source="name"   dest="text"/>
  <copyField source="author" dest="author_s"/>

  <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="text_general" class="solr.TextField" positionIncrementGap="100">
    <analyzer type="index">
      <tokenizer class="solr.StandardTokenizerFactory"/>
      <filter class="solr.StopFilterFactory" ignoreCase="true" words="stopwords.txt" />
      <filter class="solr.LowerCaseFilterFactory"/>
    </analyzer>
    <analyzer type="query">
      <tokenizer class="solr.StandardTokenizerFactory"/>
      <filter class="solr.StopFilterFactory" ignoreCase="true" words="stopwords.txt" />
      <filter class="solr.SynonymFilterFactory" synonyms="synonyms.txt" ignoreCase="true" expand="true"/>
      <filter class="solr.LowerCaseFilterFactory"/>
    </analyzer>
  </fieldType>
</schema>
```

Field Value (Index)

Joshua Bloch

Field Value (Query)

Bloch

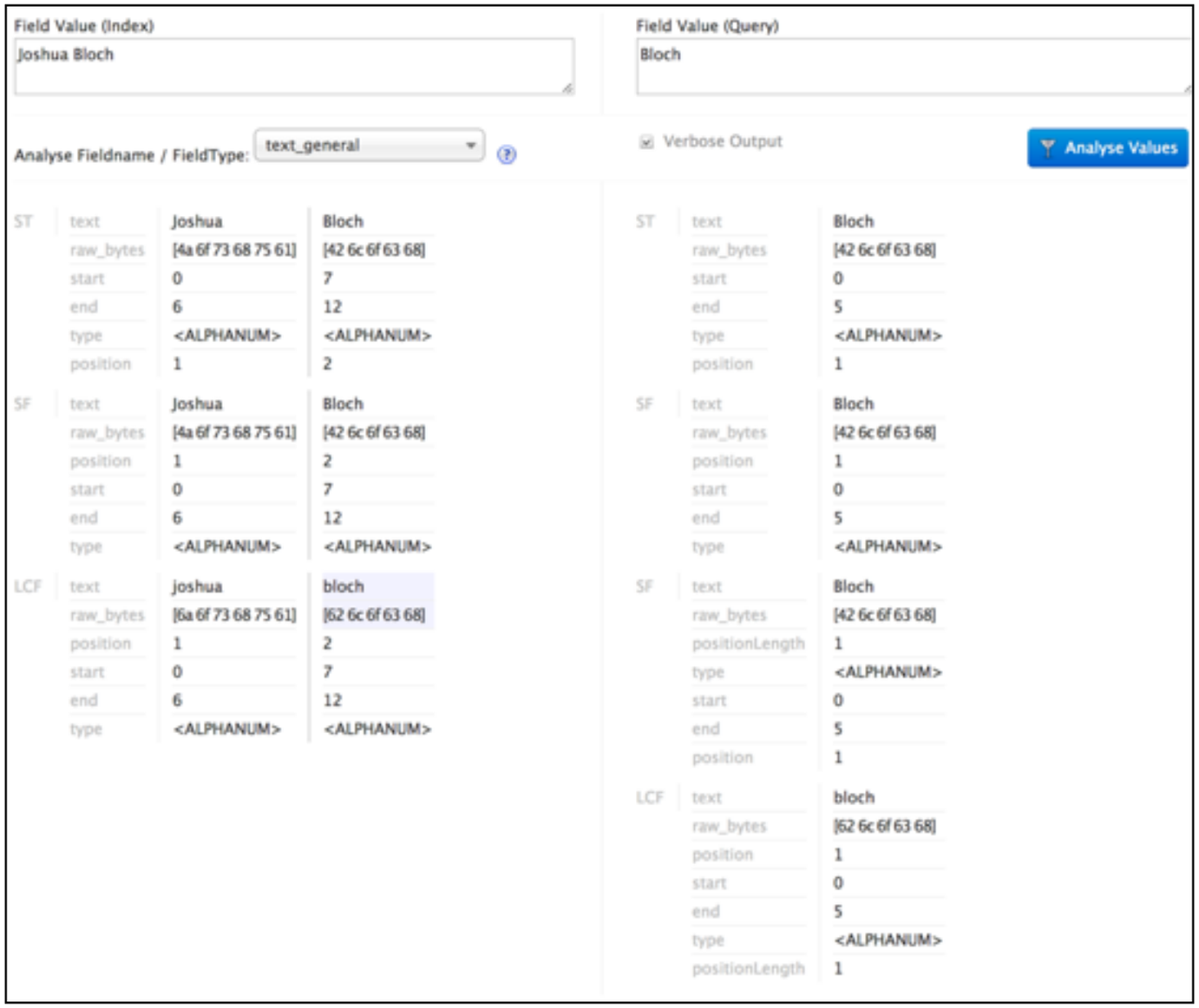
Analyse Fieldname / FieldType: string

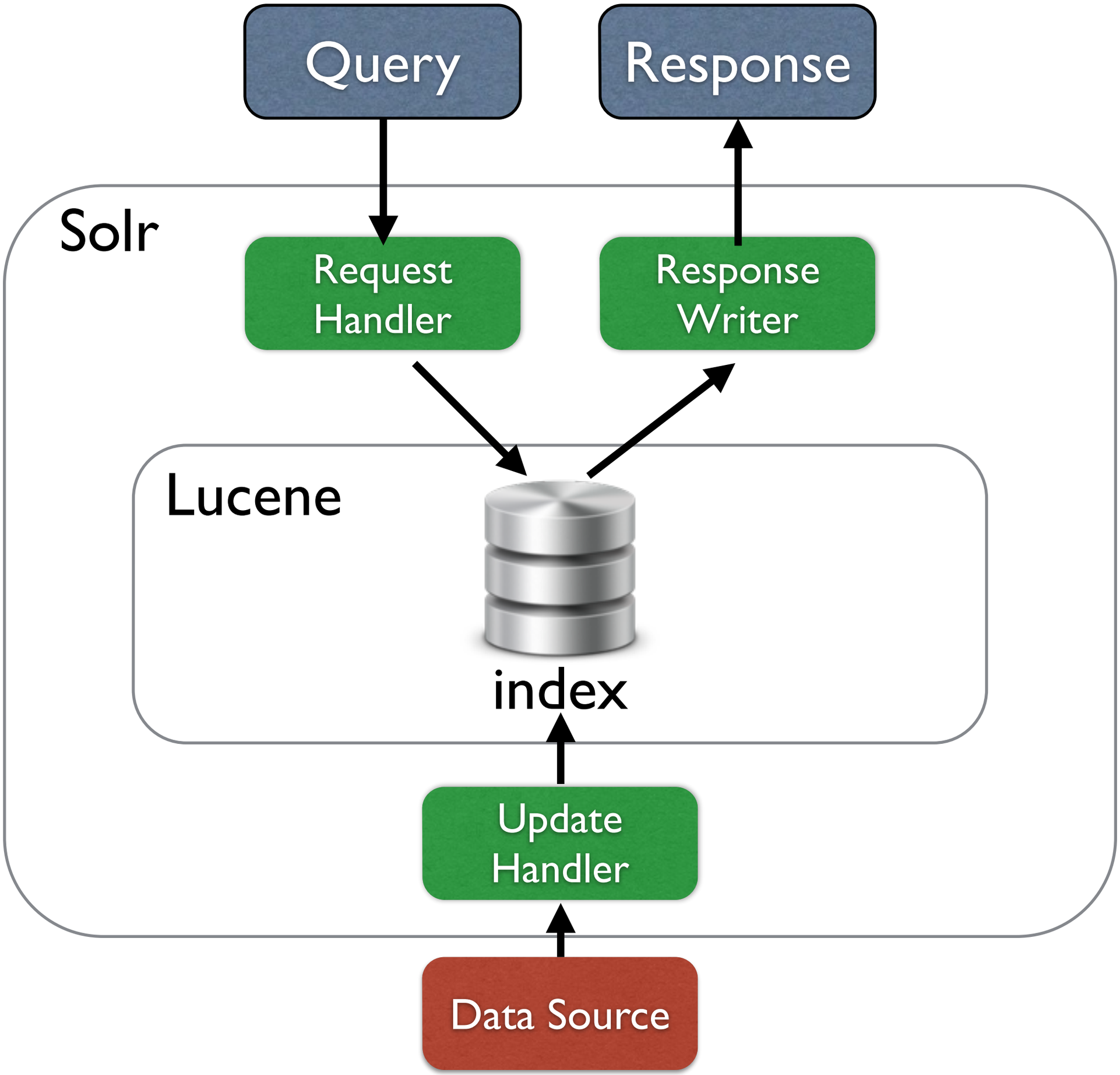
Verbose Output

Analyse Values

DA	text	Joshua Bloch
	raw_bytes	[4a 6f 73 68 75 61 20 42 6c 6f 63 68]
	start	0
	end	12
	position	1
	type	word

DA	text	Bloch
	raw_bytes	[42 6c 6f 63 68]
	start	0
	end	5
	position	1
	type	word



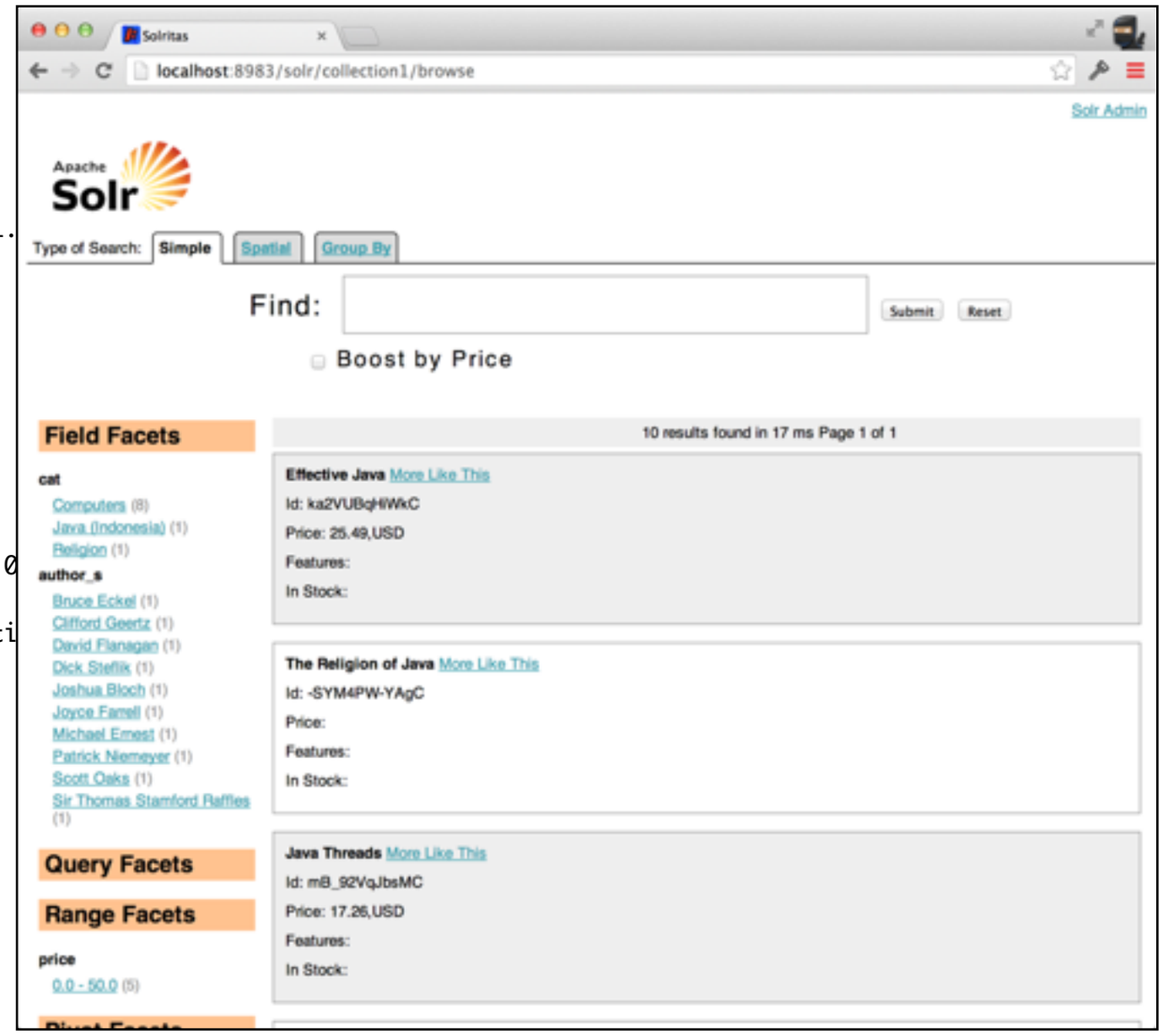

Query
Response
Solr
Request Handler
Response Writer
Lucene
index
Update Handler
Data Source

solrconfig.xml

```
<requestHandler name="/select" class="solr.SearchHandler">
  <!-- default values for query parameters can be specified, these
       will be overridden by parameters in the request
  -->
  <lst name="defaults">
    <str name="echoParams">explicit</str>
    <int name="rows">10</int>
    <str name="df">text</str>
  </lst>
</requestHandler>
```

```
<requestHandler name="/query" class="solr.SearchHandler">
  <lst name="defaults">
    <str name="echoParams">explicit</str>
    <str name="wt">json</str>
    <str name="indent">true</str>
    <str name="df">text</str>
  </lst>
</requestHandler>
```

```
<requestHandler name="/browse" class="solr.SearchHandler">
     <lst name="defaults">
       <str name="echoParams">explicit</str>

       <!-- VelocityResponseWriter settings -->
       <str name="wt">velocity</str>
       <str name="v.template">browse</str>
       <str name="v.layout">layout</str>
       <str name="title">Solritas</str>

       <!-- Query settings -->
       <str name="defType">edismax</str>
       <str name="qf">
          text^0.5 features^1.0 name^1.2 sku^1.5 id^10.0 manu^1.1 cat^1.4
          title^10.0 description^5.0 keywords^5.0 author^2.0 resourcename^1.
       </str>
       <str name="df">text</str>
       <str name="mm">100%</str>
       <str name="q.alt">*:*</str>
       <str name="rows">10</str>
       <str name="fl">*,score</str>

       <str name="mlt.qf">
         text^0.5 features^1.0 name^1.2 sku^1.5 id^10.0 manu^1.1 cat^1.4
         title^10.0 description^5.0 keywords^5.0 author^2.0 resourcename^1.0
       </str>
       <str name="mlt.fl">text,features,name,sku,id,manu,cat,title,descripti
       <int name="mlt.count">3</int>

       <!-- Highlighting defaults -->
       <str name="hl">on</str>
       <str name="hl.fl">content features title name</str>
       <str name="hl.encoder">html</str>
       <str name="hl.simple.pre">&lt;b&gt;</str>
       <str name="hl.simple.post">&lt;/b&gt;</str>
       <str name="f.title.hl.fragsize">0</str>
       <str name="f.title.hl.alternateField">title</str>
       <str name="f.name.hl.fragsize">0</str>
       <str name="f.name.hl.alternateField">name</str>
       <str name="f.content.hl.snippets">3</str>
       <str name="f.content.hl.fragsize">200</str>
       <str name="f.content.hl.alternateField">content</str>
       <str name="f.content.hl.maxAlternateFieldLength">750</str>

    <!-- append spellchecking to our list of components -->
    <arr name="last-components">
      <str>spellcheck</str>
    </arr>
</requestHandler>
```



Lab 4

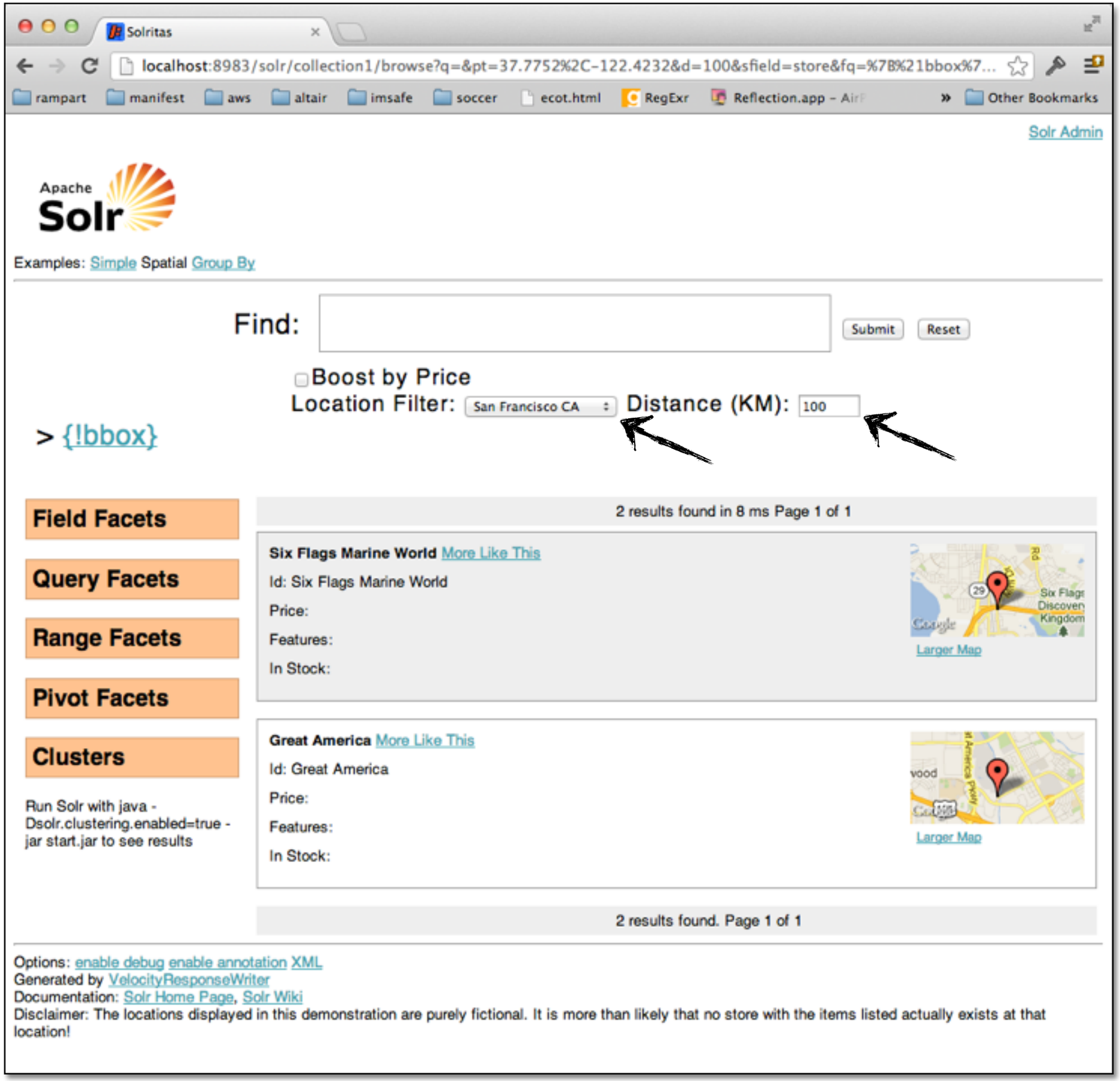
1. Add a new title field to the schema.
2. Make the title searchable and sortable
3. Restart Solr
4. Reindex
5. Search for Java in the titles
6. Sort titles containing Java in them

GEOSPATIAL

http://poiplaza.com/

```
curl /update/json --data-binary <file> -H 'Content-type:application/json'
```


```
[
    {
        "id": "Myrtle Waves Water Park",
        "name": "Myrtle Waves Water Park",
        "description": "USA-North Myrtle Beach<br />\nUS 17 Bypass at 10th Avenue<br />\n",
        "coordinates_p": "33.81673,-78.68005",
        "store": "33.81673,-78.68005",
        "source_s": "USA-Amusement & Theme Parks"
    },
    {
        "id": "Coney Island",
        "name": "Coney Island",
        "description": "USA-New York-Brooklyn<br />\n1208 Surf Ave.<br />\n+1 718-372-5159",
        "coordinates_p": "40.57546,-73.98017",
        "store": "40.57546,-73.98017",
        "source_s": "USA-Amusement & Theme Parks"
    },
    {
        "id": "Six Flags Over Georgia",
        "name": "Six Flags Over Georgia",
        "description": "USA-Austell<br />\n7561 Six Flags Pkwy<br />\n+1 770-948-9290",
        "coordinates_p": "33.77091,-84.55220",
        "store": "33.77091,-84.55220",
        "source_s": "USA-Amusement & Theme Parks"
    }
]
```



http://localhost:8983/solr/parks/browse

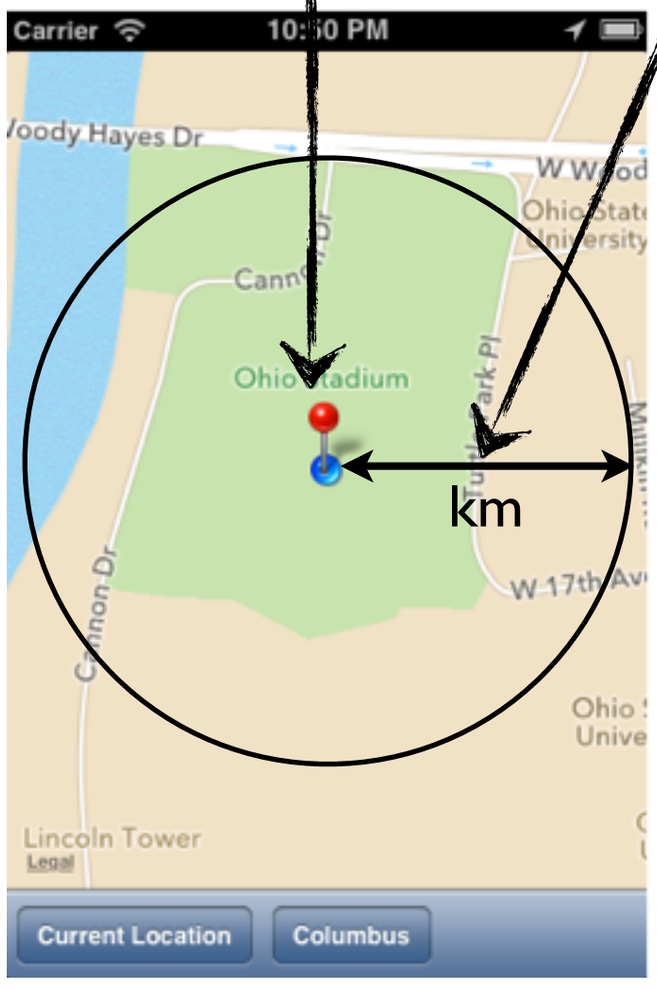
distance filters

- point distance
- bounding lat and long

point distance

```
/select?q=*:*&fq={!geofilt}&pt=37.7752,-122.4232&d=100&sfield=coordinates_p&wt=json
```

point distance filter

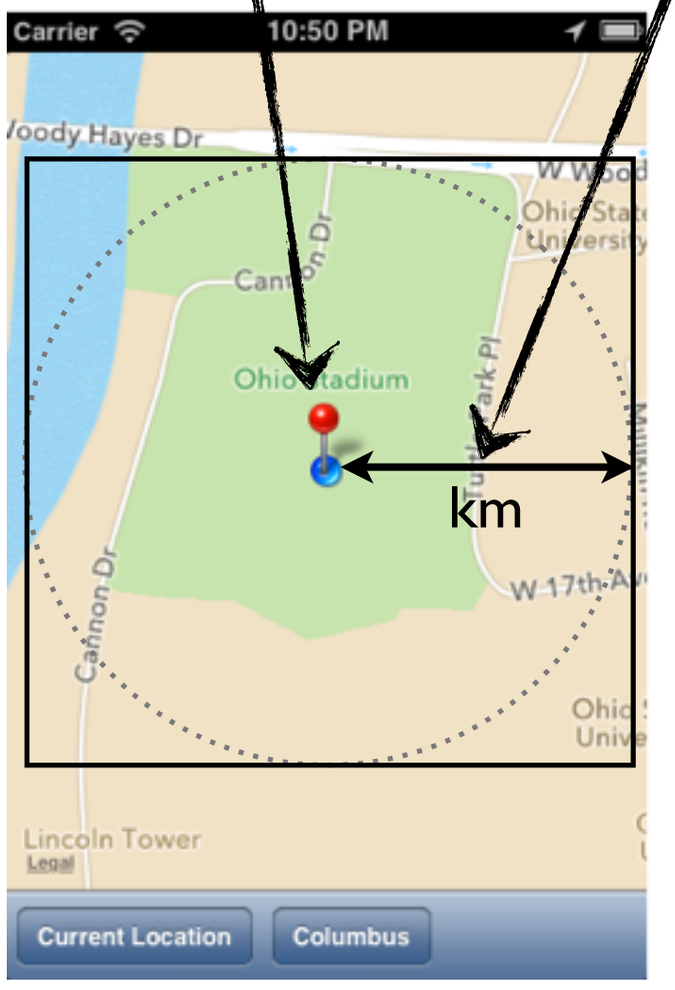



```
{
  "id": "Myrtle Waves Water Park",
  "name": "Myrtle Waves Water Park",
  "description": "USA-North Myrtle Beach-
  "coordinates_p": "33.81673,-78.68005",
  "store": "33.81673,-78.68005",
  "source_s": "USA-Amusement & Theme Parl
}
```

bounding latitude and longitude

```
/select?q=*:*&fq={!bbox}&pt=37.7752,-122.4232&d=100&sfield=coordinates_p&wt=json
```

bounding filter




```
{
  "id": "Myrtle Waves Water Park",
  "name": "Myrtle Waves Water Park",
  "description": "USA-North Myrtle Beach-
  "coordinates_p": "33.81673,-78.68005",
  "store": "33.81673,-78.68005",
  "source_s": "USA-Amusement & Theme Parl
}
```

calculate distance

```
/select?q=*:*&fq={!bbox}
&pt=37.7752,-122.4232&d=100&sfield=coordinates_p&wt=json&fl=_dist_:geodist(),name,de
scription,coordinates_p&indent=on
```


```
{
  "responseHeader":{
    "status":0,
    "QTime":0,
    "params":{
      "d":"100",
      "fl":"_dist_:geodist(),name,description,coordinates_p",
      "indent":"on",
      "q":"*:*",
      "sfield":"coordinates_p",
      "pt":"37.7752,-122.4232",
      "wt":"json",
      "fq":"{!bbox}"}},
"response":{"numFound":2,"start":0,"docs":[
    {
      "name":"Six Flags Marine World",
      "description":"USA-Vallejo<br />\n2001 Marine World Parkway<br />\n+1 707-643-6722",
      "coordinates_p":"38.14176,-122.24957",
      "_dist_":43.509483108871 28},
    {
      "name":"Great America",
      "description":"USA-Santa Clara<br />\n2401 Agnew Rd<br />\n",
      "coordinates_p":"37.39057,-121.96905",
      "_dist_":58.572205806226 02}]
}}
```

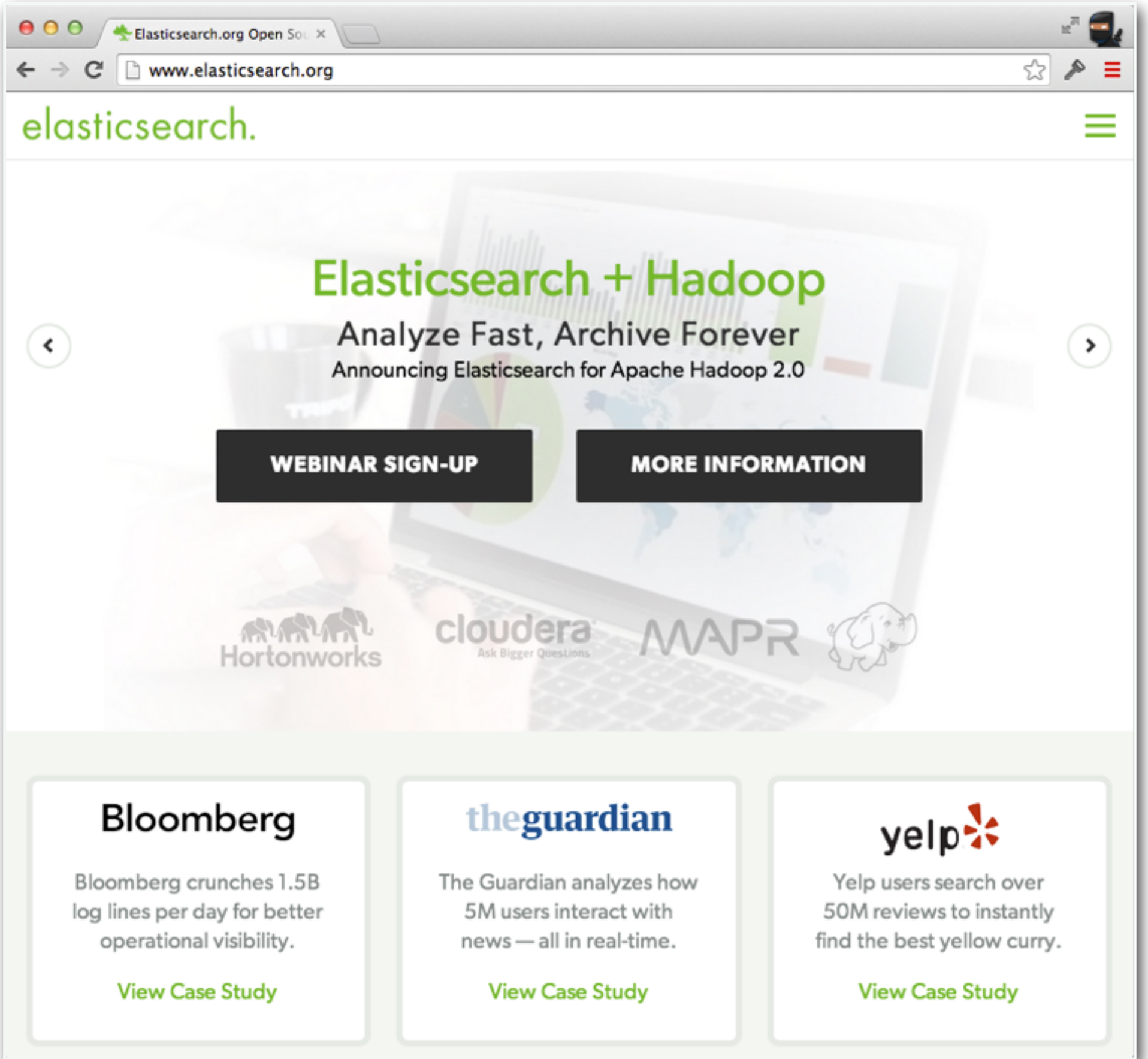
```
/select?q=*:*&fq={!bbox}
&pt=37.7752,-122.4232&d=1000&sfield=coordinates_p&wt=json&fl=_dist_:geodist(),name,d
escription,coordinates_p&indent=on&sort=geodist()+asc
```

sort by distance


```
{
  "responseHeader":{
    "status":0,
    "QTime":14,
    "params":{"d":"1000", "fl":"_dist_:geodist(),name,description,coordinates_p", "sort":"geodist() asc",
    "indent":"on","q":"*:*","sfield":"coordinates_p","pt":"37.7752,-122.4232",
    "wt":"json",
    "fq":"{!bbox}"}},
  "response":{"numFound":23,"start":0,"docs":[
      {
        "name":"Six Flags Marine World",
        "description":"USA-Vallejo<br />\n2001 Marine World Parkway<br />\n+1 707-643-6722",
        "coordinates_p":"38.14176,-122.24957",
        "_dist_":43.50948310887128},
      {
        "name":"Great America",
        "description":"USA-Santa Clara<br />\n2401 Agnew Rd<br />\n",
        "coordinates_p":"37.39057,-121.96905",
        "_dist_":58.57220580622602},
    {
      "name":"Universal Studios Hollywood",
      "description":"USA-Universal City<br />\n100 Universal City Plaza<br />\n+1 800-959-9688",
      "coordinates_p":"34.13673,-118.35590",
      "_dist_":545.5031109696113},
    {
      "name":"Santa Monica Pier",
      "description":"USA-Santa Monica<br />\n200 Santa Monica Pier<br />\n+1 310-458-8900",
      "coordinates_p":"34.01036,-118.49612",
      "_dist_":547.9619130243483},
    {
      "name":"Raging Waters",
      "description":"USA-San Dimas<br />\n111 Raging Waters Drive<br />\n+1 909-802-2200",
      "coordinates_p":"34.08565,-117.81186",
      "_dist_":583.5368017552456},
    {
      "name":"Knott's Berry Farm/Soak City",
      "description":"USA-Buena Park<br />\n8039 Beach Blvd.<br />\n+1 714-220-5200",
      "coordinates_p":"33.84550,-117.99810",
      "_dist_":591.5876572669723},
    {
      "name":"Disneyland",
      "description":"USA-Anaheim<br />\n700 W Ball Rd<br />\n+1 714-781-4565",
      "coordinates_p":"33.81786,-117.91846",
      "_dist_":598.7499877946933},
```

Lab 5

1. Clear the current index
2. Load the amusement park data
3. Search for amusement parks near (100 meters) of Orlando Florida (28.405080, -81.579433)



http://www.elasticsearch.org/

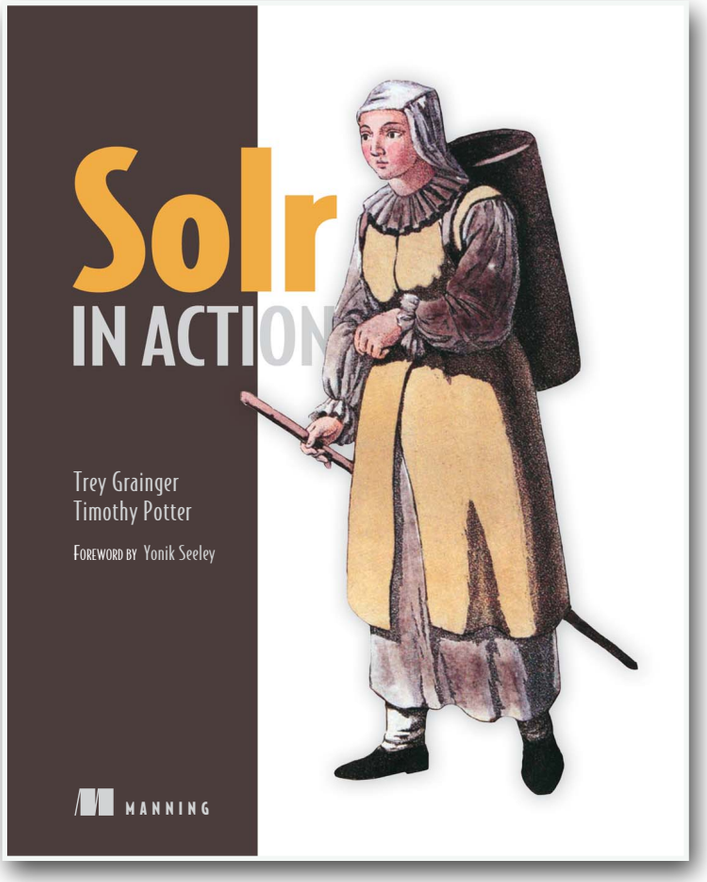
ELK

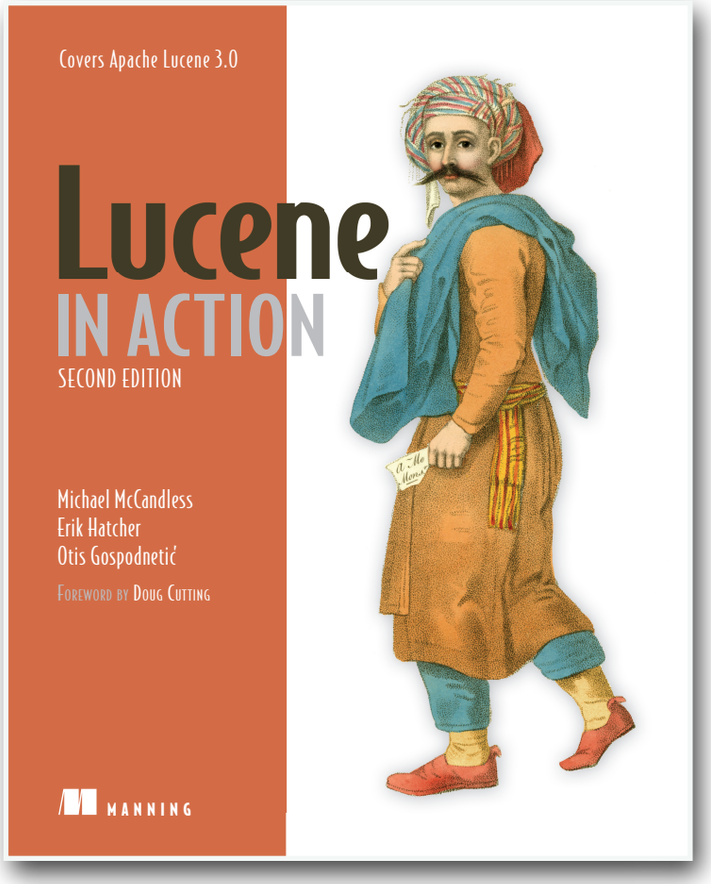
collection
and processing

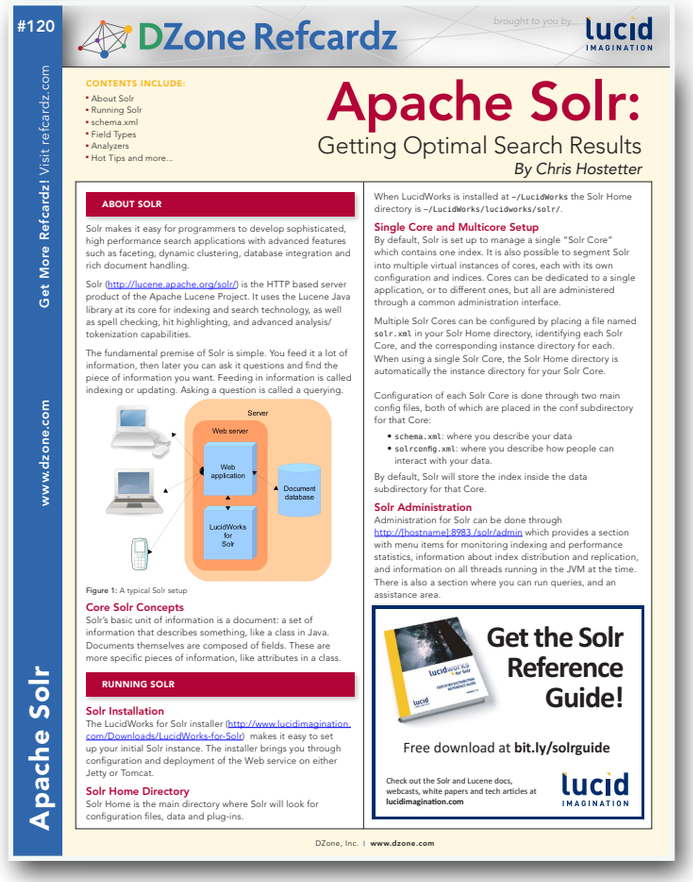
indexing
and searching

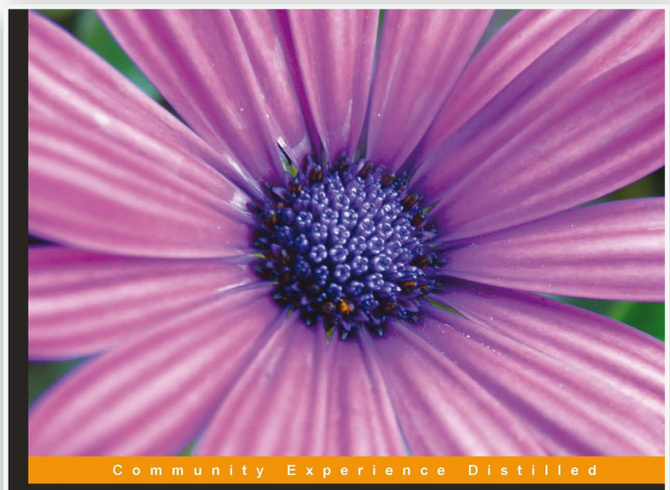
user interface

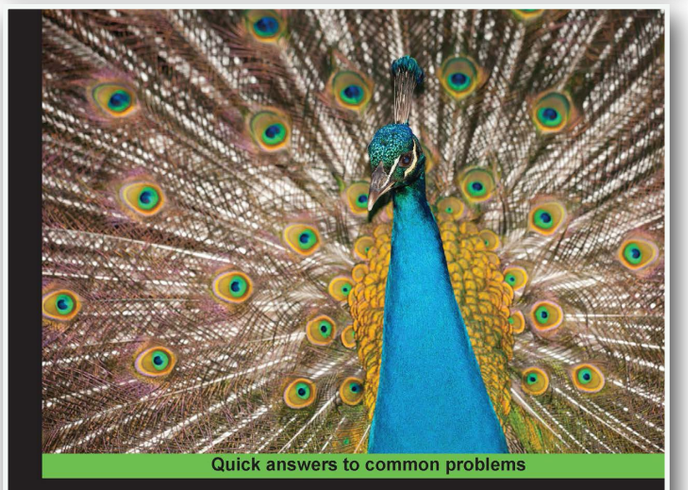
input | codecs | filters | outputs


Solr
IN ACTION
Trey Grainger
Timothy Potter
Foreword by Yonik Seeley
MANNING

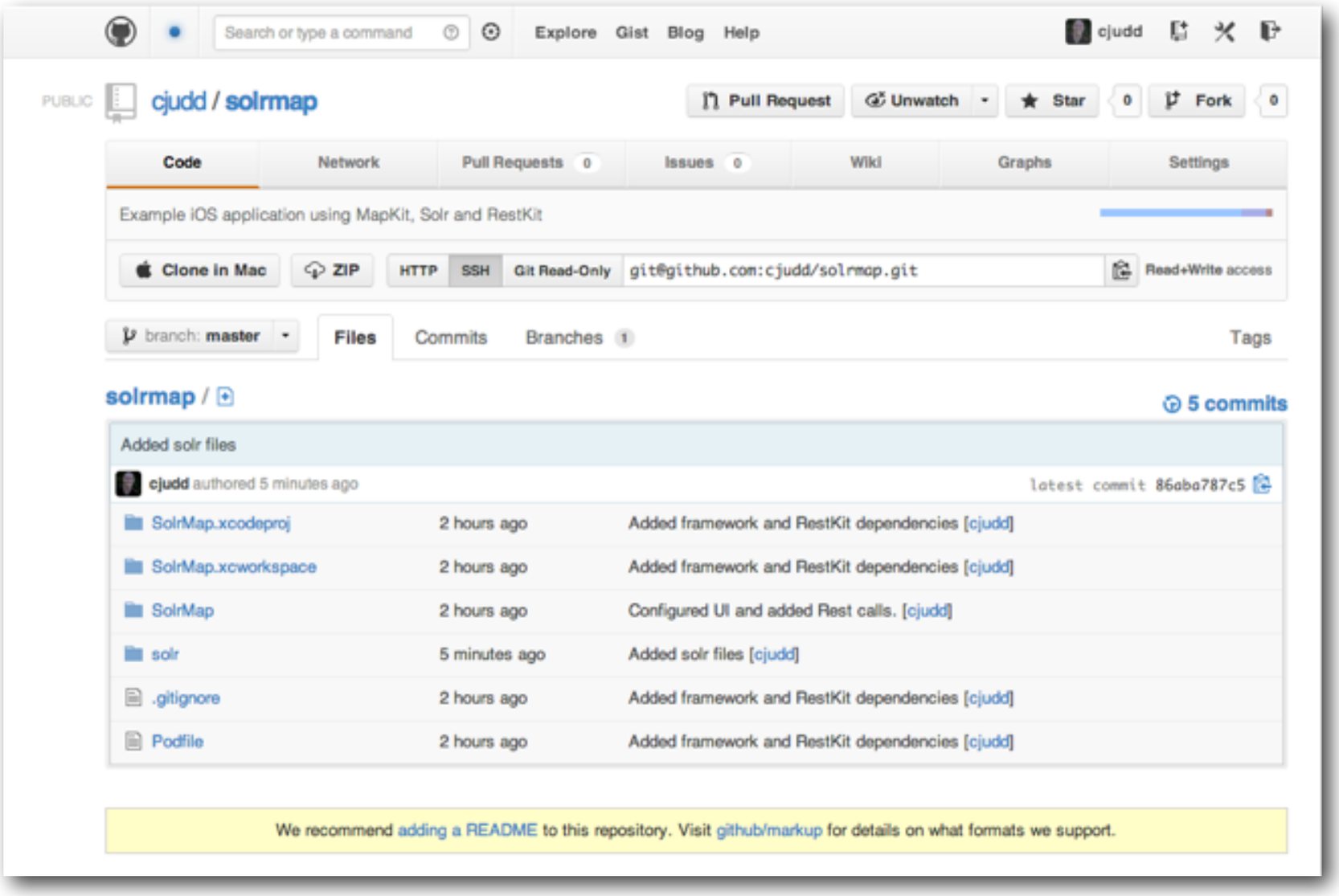

Covers Apache Lucene 3.0
Lucene
IN ACTION
SECOND EDITION
Michael McCandless
Erik Hatcher
Otis Gospodnetić
Foreword by Doug Cutting
MANNING

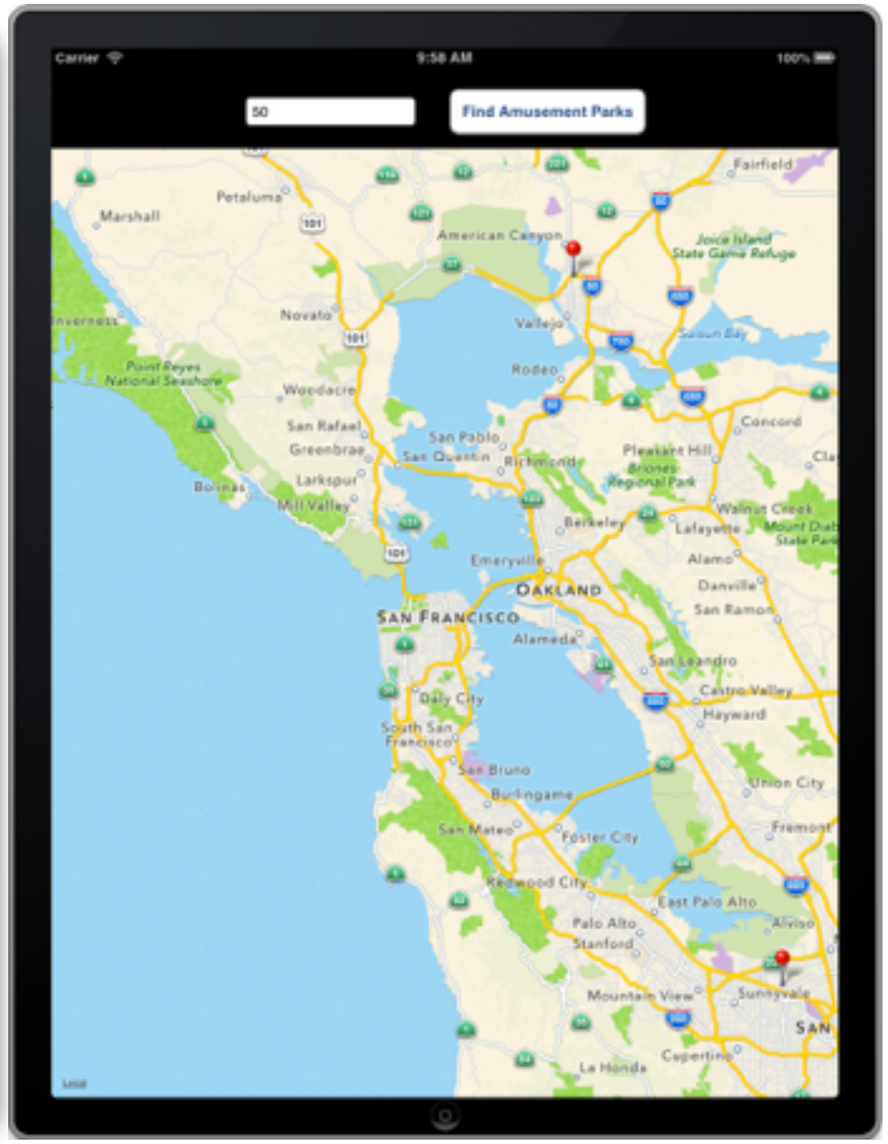

#120
DZone Refcardz
brought to you by: lucid IMAGINATION
Get More Refcardz! Visit refcardz.com
www.dzone.com
Apache Solr
CONTENTS INCLUDE:
• About Solr
• Running Solr
• schema.xml
• Field Types
• Analyzers
• Hot Tips and more...
Apache Solr:
Getting Optimal Search Results
By Chris Hostetter
ABOUT SOLR
Solr makes it easy for programmers to develop sophisticated, high performance search applications with advanced features such as faceting, dynamic clustering, database integration and rich document handling.
Solr (http://lucene.apache.org/solr/) is the HTTP based server product of the Apache Lucene Project. It uses the Lucene Java library at its core for indexing and search technology, as well as spell checking, hit highlighting, and advanced analysis/tokenization capabilities.
The fundamental premise of Solr is simple. You feed it a lot of information, then later you can ask it questions and find the piece of information you want. Feeding in information is called indexing or updating. Asking a question is called a querying.
Server
Web server
Web application
Document database
LucidWorks for Solr
Figure 1: A typical Solr setup
Core Solr Concepts
Solr's basic unit of information is a document: a set of information that describes something, like a class in Java. Documents themselves are composed of fields. These are more specific pieces of information, like attributes in a class.
RUNNING SOLR
Solr Installation
The LucidWorks for Solr installer (http://www.lucidimagination.com/Downloads/LucidWorks-for-Solr) makes it easy to set up your initial Solr instance. The installer brings you through configuration and deployment of the Web service on either Jetty or Tomcat.
Solr Home Directory
Solr Home is the main directory where Solr will look for configuration files, data and plug-ins.
When LucidWorks is installed at ~/LucidWorks the Solr Home directory is ~/LucidWorks/lucidworks/solr/.
Single Core and Multicore Setup
By default, Solr is set up to manage a single "Solr Core" which contains one index. It is also possible to segment Solr into multiple virtual instances of cores, each with its own configuration and indices. Cores can be dedicated to a single application, or to different ones, but all are administered through a common administration interface.
Multiple Solr Cores can be configured by placing a file named solr.xml in your Solr Home directory, identifying each Solr Core, and the corresponding instance directory for each. When using a single Solr Core, the Solr Home directory is automatically the instance directory for your Solr Core.
Configuration of each Solr Core is done through two main config files, both of which are placed in the conf subdirectory for that Core:
• schema.xml: where you describe your data
• solrconfig.xml: where you describe how people can interact with your data.
By default, Solr will store the index inside the data subdirectory for that Core.
Solr Administration
Administration for Solr can be done through http://[hostname]:8983/solr/admin which provides a section with menu items for monitoring indexing and performance statistics, information about index distribution and replication, and information on all threads running in the JVM at the time. There is also a section where you can run queries, and an assistance area.
Get the Solr Reference Guide!
Free download at bit.ly/solrguide
Check out the Solr and Lucene docs, webcasts, white papers and tech articles at lucidimagination.com
lucid IMAGINATION
DZone, Inc. | www.dzone.com


Community Experience Distilled
Apache Solr 3 Enterprise Search Server
Enhance your search with faceted navigation, result highlighting, relevancy ranked sorting, and more
David Smiley
Eric Pugh
[PACKT] open source
PUBLISHING


Quick answers to common problems
Apache Solr 4 Cookbook
Over 100 recipes to make Apache Solr faster, more reliable, and return better results
Rafał Kuć
[PACKT] open source
PUBLISHING

MapKit, Solr and RestKit





https://github.com/cjudd/solrmap

Christopher M. Judd

Manifest Solutions

CTO and Partner

email: cjudd@juddsolutions.com

web: www.juddsolutions.com

blog: juddsolutions.blogspot.com

twitter: javajudd