Apache Solr

Yonik Seeley
yonik@apache.org

29 June 2006
Dublin, Ireland
History

- Search for a replacement search platform
  - commercial: high license fees
  - open-source: no full solutions
- CNET grants code to Apache, Solr enters Incubator 17 Jan 2006
- Solr is a Lucene sub-project
- Users: CNET Reviews, CNET Channel, shopper.com, news.com, nines.org, krugle.com, oodle.com, booklooker.de
Lucene Refresher

- Lucene is a full-text search library
- Add documents to an index via IndexWriter
  - A document is a collection of fields
  - No config files, dynamic field typing
  - Flexible text analysis - tokenizers, filters
- Search for documents via IndexSearcher
  \[
  \text{Hits} = \text{search(\text{Query}, \text{Filter}, \text{Sort}, \text{topN})}
  \]
- Scoring: \( \text{tf} \times \text{idf} \times \text{lengthNorm} \)
What Is Solr

• A full text search server based on Lucene
• XML/HTTP Interfaces
• Loose Schema to define types and fields
• Web Administration Interface
• Extensive Caching
• Index Replication
• Extensible Open Architecture
• Written in Java5, deployable as a WAR
Architecture

Admin Interface
- Standard Request Handler
- Disjunction Max Request Handler
- Custom Request Handler
- XML Response Writer

HTTP Request Servlet
- Config
- Schema
- Solr Core
- Concurrency

Update Servlet
- Caching
- Update Handler

Lucene

Replication
Adding Documents

HTTP POST to /update

<add><doc boost="2">
  <field name="article">05991</field>
  <field name="title">Apache Solr</field>
  <field name="subject">An intro…</field>
  <field name="category">search</field>
  <field name="category">lucene</field>
  <field name="body">Solr is a full…</field>
</doc></add>
Deleting Documents

- Delete by Id
  <delete><id>05591</id></delete>

- Delete by Query (multiple documents)
  <delete>
    <query>manufacturer:microsoft</query>
  </delete>
Commit

- `<commit/>` makes changes visible
  - closes IndexWriter
  - removes duplicates
  - opens new IndexSearcher
    - newSearcher/firstSearcher events
    - cache warming
    - “register” the new IndexSearcher

- `<optimize/>` same as commit, merges all index segments.
Default Query Syntax

**Lucene Query Syntax** [; sort specification]

1. `mission impossible; releaseDate desc`
2. `+mission +impossible -actor:cruise`
3. “mission impossible” -actor:cruise
4. `title:spiderman^10 description:spiderman`
5. `description:“spiderman movie”~10`
6. `+HDTV +weight:[0 TO 100]`
7. Wildcard queries: `te?t, te*t, test*`
# Default Parameters

Query Arguments for HTTP GET/POST to `/select`

<table>
<thead>
<tr>
<th>param</th>
<th>default</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td></td>
<td>The query</td>
</tr>
<tr>
<td>start</td>
<td>0</td>
<td>Offset into the list of matches</td>
</tr>
<tr>
<td>rows</td>
<td>10</td>
<td>Number of documents to return</td>
</tr>
<tr>
<td>fl</td>
<td>*</td>
<td>Stored fields to return</td>
</tr>
<tr>
<td>qt</td>
<td>standard</td>
<td>Query type; maps to query handler</td>
</tr>
<tr>
<td>df</td>
<td>(schema)</td>
<td>Default field to search</td>
</tr>
</tbody>
</table>
Search Results

http://localhost:8983/solr/select?q=video&start=0&rows=2&fl=name,price

<response>
  <responseHeader>
    <status>0</status>
    <QTime>1</QTime></responseHeader>
  <result numFound="16173" start="0">
    <doc>
      <str name="name">Apple 60 GB iPod with Video</str>
      <float name="price">399.0</float>
    </doc>
    <doc>
      <str name="name">ASUS Extreme N7800GTX/2DHTV</str>
      <float name="price">479.95</float>
    </doc>
  </result>
</response>
Caching

IndexSearcher’s view of an index is fixed
  - Aggressive caching possible
  - Consistency for multi-query requests

filterCache - unordered set of document ids matching a query

resultCache - ordered subset of document ids matching a query

documentCache - the stored fields of documents

userCaches - application specific, custom query handlers
Warming for Speed

- Lucene IndexReader warming
  - field norms, FieldCache, tii - the term index
- Static Cache warming
  - Configurable static requests to warm new Searchers
- Smart Cache Warming (autowarming)
  - Using MRU items in the current cache to pre-populate the new cache
- Warming in parallel with live requests
Smart Cache Warming

On-Deck Solr IndexSearcher

User Cache
Filter Cache
Result Cache
Doc Cache

Regenerator

Warming Requests
Live Requests

Request Handler

Registered Solr IndexSearcher

User Cache
Filter Cache
Result Cache
Doc Cache

Field Cache
Field Norms

1. Auto-warming
2. 2
3. 3

Autowarming - warm n MRU cache keys w/ new Searcher
Schema

- Lucene has no notion of a schema
  - Sorting - string vs. numeric
  - Ranges - val:42 included in val:[1 TO 5] ?
  - Lucene QueryParser has date-range support, but must guess.
- Defines fields, their types, properties
- Defines unique key field, default search field, Similarity implementation
Field Definitions

• Field Attributes: name, type, indexed, stored, multiValued, omitNorms
  <field name="id" type="string" indexed="true" stored="true"/>
  <field name="sku" type="textTight" indexed="true" stored="true"/>
  <field name="name" type="text" indexed="true" stored="true"/>
  <field name="reviews" type="text" indexed="true" stored="false"/>
  <field name="category" type="text_ws" indexed="true" stored="true" multiValued="true"/>

• Dynamic Fields, in the spirit of Lucene!
  <dynamicField name="*_i" type="sint" indexed="true" stored="true"/>
  <dynamicField name="*_s" type="string" indexed="true" stored="true"/>
  <dynamicField name="*_t" type="text" indexed="true" stored="true"/>
Search Relevancy

Document Analysis

PowerShot SD 500

WhitespaceTokenizer

PowerShot SD 500

WordDelimiterFilter catenateWords=1

Power Shot SD 500

LowercaseFilter

power shot sd 500

Query Analysis

power-shot sd500

WhitespaceTokenizer

power-shot sd500

WordDelimiterFilter catenateWords=0

power shot sd 500

LowercaseFilter

power shot sd 500

A Match!
Configuring Relevancy

```xml
<fieldtype name="text" class="solr.TextField">
  <analyzer>
    <tokenizer class="solr.WhitespaceTokenizerFactory"/>
    <filter class="solr.LowerCaseFilterFactory"/>
    <filter class="solr.SynonymFilterFactory" synonyms="synonyms.txt"/>
    <filter class="solr.StopFilterFactory" words="stopwords.txt"/>
    <filter class="solr.EnglishPorterFilterFactory" protected="protwords.txt"/>
  </analyzer>
</fieldtype>
```
copyField

- Copies one field to another at index time
- Use case: Analyze same field different ways
  - Copy into a field with a different analyzer
  - Boost exact-case, exact-punctuation matches
  - Language translations, thesaurus, soundex

```xml
<field name="title" type="text"/>
<field name="title_exact" type="text_exact" stored="false"/>
<copyField source="title" dest="title_exact"/>
```

- Use case: Index multiple fields into single searchable field
High Availability

Load Balancer

Appservers

Solr Searchers

Solr Master

DB Updater

Dynamic HTML Generation

HTTP search requests

Index Replication

admin queries

updates

admin terminal

Updater

DB

ApacheCon Europe 06
Replication

1. hard links

2. hard links

3. rsync

4. mv dir

Lucene index segments

solr/data/index

solr/data/index

solr/data/snapshot-2006062950000

solr/data/snapshot-2006062950000-WIP

Master

Searcher

ApacheCon Europe 06

20
Faceted Browsing Example

You found 1045 items for System type: Budget desktop system
Too few results? Click a link above to remove that filter, or remove all filters.

Find by price
- Less than $400 (76)
- $400 to $699 (337)
- $700 to $999 (468)
- $1000 to $1299 (5)

Find by manufacturer
- Dell, Inc. (43)
- Lenovo (490)
- HP (342)
- Acer America Corp. (28)
- Cyberpower Inc (22)
- See all manufacturers

Find by processor manufacturer
- Intel (804)
- AMD (122)
- Motorola (1)

Or find by
- Clock speed
- Graphics processor
- RAM installed
- Hard drive size
- OS provided
- See all

Sort by: Product name | Lowest price | Editors' rating | Review date

Dell Dimension B110 Desktop Computer for Home (Cel-D 2.53GHz/160GB/512MB)
Dell's entry-level Dimension B110 series features aging technology and a dated design, but its members will suffice as second PCs for basic tasks.
Specs: Celeron D (2.53 GHz), 512 MB, 160 GB, 15 in, Microsoft Windows XP Home Edition

Add to my products New! What is this?

$479
at 1 store

Check prices

Dell Dimension B110 Desktop Computer for Home (Cel-D 2.53GHz/80GB/256MB)
Dell's entry-level Dimension B110 series features aging

$349
at 1 store
Faceted Browsing

Search(Query, Filter[], Sort, offset, n)

DocList

DocSet

Intersection Size:
- computer_type: PC
- memory: [1GB TO *]
- proc_manu: Intel
- proc_manu: AMD
- price: [0 TO 500]
- price: [500 TO 1000]
- manu: Dell
- manu: HP
- manu: Lenovo

Query Response

Size:
- = 594
- = 382
- = 247
- = 689
- = 104
- = 92
- = 75
Web Admin Interface

- Show Config, Schema, Distribution info
- Query Interface
- Statistics
  - Caches: lookups, hits, hitratio, inserts, evictions, size
  - RequestHandlers: requests, errors
  - UpdateHandler: adds, deletes, commits, optimizes
  - IndexReader, open-time, index-version, numDocs, maxDocs,
- Analysis Debugger
  - Shows tokens after each Analyzer stage
  - Shows token matches for query vs index
Selling Points

- Fast
- Powerful & Configurable
- High Relevancy
- Mature Product
- Same features as software costing $$$
- Leverage Community
  - Lucene committers, IR experts
  - Free consulting: shared problems & solutions
Where are we going?

- OOTB Simple Faceted Browsing
- Automatic Database Indexing
- Federated Search
  - HA with failover
- Alternate output formats (JSON, Ruby)
- Highlighter integration
- Spellchecker
- Alternate APIs (Google Data, OpenSearch)
Resources

- **WWW**
  - [http://incubator.apache.org/solr](http://incubator.apache.org/solr)

- **Mailing Lists**
  - solr-user-subscribe@lucene.apache.org
  - solr-dev-subscribe@lucene.apache.org