



Rapid Prototyping Search Applications with Solr



Presented by Erik Hatcher
Technical Staff, Lucid Imagination

Why prototype?

- Demonstrate Solr can handle your needs
- Mitigate risk, learn the unknown
- The User Interface is the app
- It's quick, easy, AND FUN!



LucidWorks for Solr

- **Great starting point**
- **Built-in and pre-configured:**
 - ▼ Clustering
 - ▼ Carrot2
 - ▼ Search UI
 - ▼ Solritas (VelocityResponseWriter)
 - ▼ Server includes root context, handy for serving static files
 - ▼ Better stemming
 - ▼ KStem
 - ▼ choice of Tomcat or Jetty

The Requirement

Make your *<Big Enterprise Content Repository>* searchable

- ▼ PDF, Word, PowerPoint, HTML,...
- ▼ Accessed through proprietary API



Do the simplest next step towards the goal

- ▶ Let's just index a PDF file



File indexing first attempt

```
curl "http://localhost:8983  
/solr/update/extract  
?stream.file=/docs/file.pdf"
```

Document [null] missing required field: id

```
from schema.xml  
<field name="id" type="string"  
      indexed="true" stored="true"  
      required="true" />  
  
<uniqueKey>id</uniqueKey>
```

Unique Key

- **Practically all Solr-based applications use a unique key for each document**
- **Required to "update" a document, and some components need it**
- **Determining a unique key scheme:**
 - ▼ May be obvious
 - ▼ a DB primary key or URL
 - ▼ May involve a new scheme, especially with multiple data sources
 - ▼ perhaps prefix data-source specific id's with the data source code:
 - ▼ <data-source>-<document-id-within-datasource>
 - ▼ Examples: product-1234, article-1234

Unique identifier

```
curl "http://localhost:8983  
/solr/update/extract  
?stream.file=/docs/file.pdf  
&literal.id=/docs/file.pdf "
```

```
<response>  
  <lst name="responseHeader">  
    <int name="status">0</int>  
    <int name="QTime">1838</int>  
  </lst>  
</response>
```

http://localhost:8983/solr/itas

Pronounced: so-LAIR-uh-toss

Query: <input type="text"/>	
1 results found in 22 ms	
Facets cat	last_modified : Fri Jun 04 14:09:57 EDT 2010
Clusters • Other Topics	content_type : application/pdf
	id : /docs/file.pdf
	score : 1.0
1 results found. Page 1 of 1	
Generated by VelocityResponseWriter	

- **Pronounced: so-LAIR-uh-toss**
- **Celeritas is a Latin word, translated as "swiftness" or "speed". It is often given as the origin of the symbol c, the universal notation for the speed of light - <http://en.wikipedia.org/wiki/Celeritas>**
- **VelocityResponseWriter - simply passes the Solr response through the Apache Velocity templating engine**
- **<http://wiki.apache.org/solr/VelocityResponseWriter>**

Keeping it Clean

- **Customize the schema**
 - ▼ Remove example fields
- **Make URLs domain-specific**
 - ▼ Remove unused/example request handlers
 - ▼ Add custom handlers with your defaults
 - ▼ Note: tinkering with URLs requires client / template changes too
 - ▼ specifically in `browse.vm` and `VM_global_library.vm`
- **Make a habit of tidying up after each step!**

Specific schema changes

Added stored body field (schema.xml)

```
+ <field name="body" type="text" indexed="true"
  stored="true" />
```

Copy all fields into catch-all "text" field (schema.xml)

```
+ <copyField source="*" dest="text" />
```

Adjusted /update/extract to body field (solrconfig.xml)

```
<!-- All the main content goes into "text"... if you need to
return
the extracted text or do highlighting, use a stored field. -->
```

```
<str name="fmap.content">text</str>+
<str name="fmap.content">body</str>
```

Get rid of the /itas!

```

<requestHandler name="/browse" class="solr.SearchHandler">
  <lst name="defaults">
    <!-- UI settings -->
    <str name="wt">velocity</str>
    <str name="v.template">browse</str>
    <str name="v.layout">layout</str>
    <str name="title">My File Search Prototype</str>

    <!-- results details -->
    <str name="rows">10</str>
    <str name="fl">id, content_type, last_modified, score</str>

    <!-- query parsing -->
    <str name="defType">lucene</str>
    <str name="q">*:*</str>

    <!-- faceting -->
    <str name="facet">on</str>
    <str name="facet.field">content_type</str>
    <str name="facet.mincount">1</str>
  </lst>
</requestHandler>

```

Faceting

Facets

content_type

- [application/pdf](#) (1)

http://localhost:8983/solr/browse

```
<requestHandler name="/browse" class="solr.SearchHandler">
  <lst name="defaults">
    <!-- UI settings -->
    <str name="wt">velocity</str>
    <str name="v.template">browse</str>
    <str name="v.layout">layout</str>
    <str name="title">My File Search Prototype</str>

    <!-- results details -->
    <str name="rows">10</str>
    <str name="fl">id,content_type,last_modified,score</str>

    <!-- query parsing -->
    <str name="defType">lucene</str>
    <str name="q">*:*</str>

    <!-- faceting -->
    <str name="facet">on</str>
    <str name="facet.field">content_type</str>
    <str name="facet.mincount">1</str>
  </lst>
</requestHandler>
```

Changing Solr's config

Prototyping peace of mind:

- ▼ Backup original files :)
- ▼ Stop LucidWorks for Solr (ctrl-c)
- ▼ Delete index (rm -Rf lucidworks/solr/data)
 - ▼ Always be able to reindex from scratch!
- ▼ Restart LucidWorks for Solr (./start.sh)
- ▼ Reindex

Customizing results display

velocity/hit.vm

```
<div class="result-document">
  <b>$doc.getFieldValue('id')</b>
  <p>Last modified:
    $!doc.getFieldValue('last_modified')
  </p>
```

...

Leave default debugging bit there, you'll want it later

/Users/erikhatcher/dev/lucid/presentations/webinars/prototyping-june-2010/docs/ACEU09-Lucene Case Studies.pdf

Last modified: Fri Jun 04 11:09:57 PDT 2010

last_modified unknown

```
#if ( $doc. get Fi el dVal ue( ' l a s t _ m o d i f i e d' ) )  
<p>Last  m o d i f i e d:  
$doc. get Fi el dVal ue( ' l a s t _ m o d i f i e d' ) </ p>#end
```

Hyperlinking to files

```
<a href="file://$doc.getFieldValue('id') ">  
  $doc.getFieldValue('id')  
</a>
```

Note: responsible browsers disallow file:// links from working here (unless otherwise configured), though copying and pasting the link should work in a new window.

Highlighting search terms

add to solrconfig.xml

```

<requestHandler name="/browse" class="solr.SearchHandler">
  <lst name="defaults">
    ...
    <!-- highlighting -->
    <str name="hl">on</str>
    <str name="hl.fl">body</str>
    <str name="hl.snippets">3</str>

  </lst>
</requestHandler>
  
```

Highlighting display

```

i n hit.vm
<p>
foreach($fragment in
$response.response.highlighting.get($doc.getFieldValue('id')).body)
    ... $fragment ...
#end</p>

```

</Users/erikhatcher/dev/lucid/presentations/webinars/prototyping-june-2010/docs/eurocon10-Rapid Prototyping with Solr.ppt>

... better, and finally showcase your data in a flexible *search* user interface. We'll see how to rapidly ...
 ... time left to outline the next steps in developing your *search* application and taking it to production ...
 ... Works for Solr great starting point built-in and pre-configured: Clustering Carrot2 Search UI SolrItas
 (Velocity ...

Adding spell checking

schema.xml changes

- ▼ Add textSpell field type to schema.xml
- ▼ Add spell field, of type textSpell
- ▼ copyField desired fields into spell field

solrconfig.xml changes

- ▼ change the spellchecker field name to "spell"
- ▼ set spellchecker buildOnCommit to true
- ▼ add spellcheck component and options to handler

Stop, delete data/ directory, restart, reindex

Add spell check suggestions to UI

Spellcheck config

schema.xml

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

+ <field name="spell" type="textSpell" indexed="true" stored="false"
multiValued="true"/>
+ <copyField source="body" dest="spell"/>
```

solrconfig.xml

```
-<str name="field">name</str>
+<str name="field">spell</str>
+<str name="buildOnCommit">true</str>

+   <!-- spellchecking -->
+   <str name="spellcheck">on</str>
+   <str name="spellcheck.collate">true</str>

+   <arr name="last-components">
+     <str>spellcheck</str>
+   </arr>
```

Did you mean...?

Query: user interfarcas

Did you mean [user interfaces](#)?

Added to browse.vm

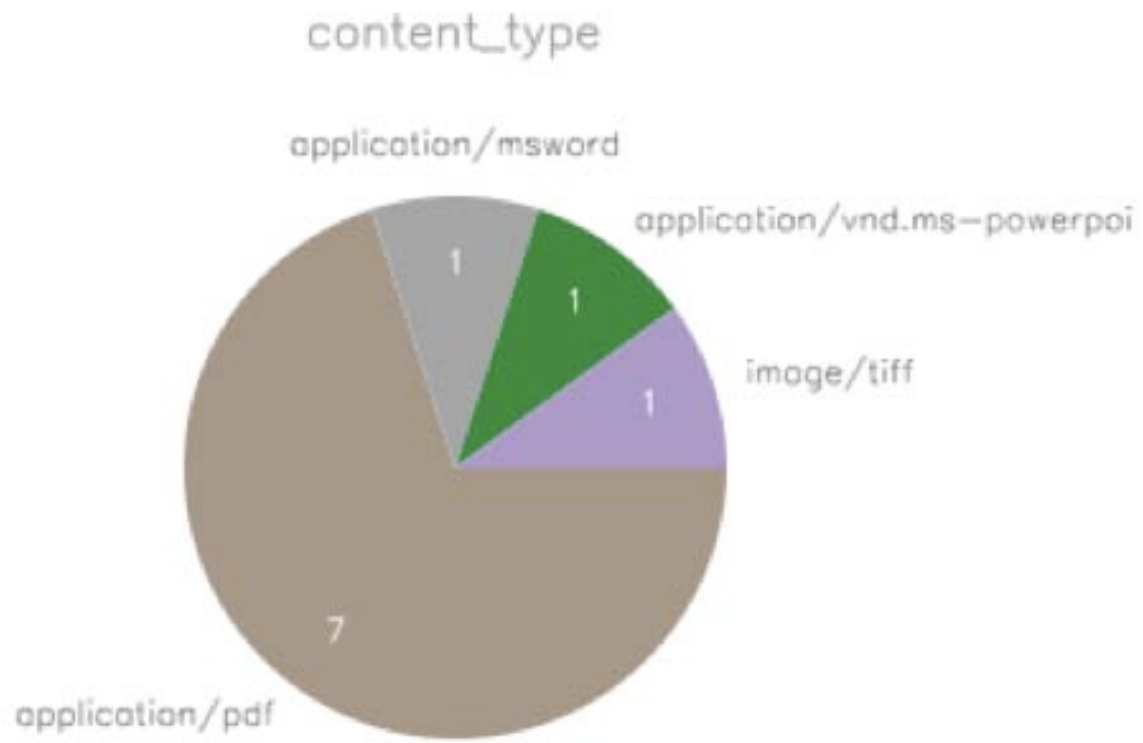
```
#if($response.response.spellcheck.suggestions.size() > 0)
```

Did you mean <a

```
href="/solr/browse?q=$esc.url($response.response.spellcheck.suggestions.collation)">$response.response.spellcheck.suggestions.collation</a>?
```

```
#end
```

Dessert: Pie



How the chart came to life

- **Found simple JavaScript chart package:**
<http://www.jscharts.com>
- **Looked at an example**
- **Downloaded**
 - ▶ placed jschart.js in
~/LucidWorks/lucidworks/jetty/webapps/root/scripts/
- **Integrated**

JSChart integration

added to layout.vm

```
<script type="text/javascript"
src="/scripts/jscharts.js"></script>
```

conf/velocity/jschart.vm

```
#set($facet_field=$request.params.get('facet.field'))
#set($chart_type=$request.params.get('jschart.type'))
#set($facets=$response.response.facet_counts.facet_fields.get($facet_field))
<div id="jschart_${chart_type}_${facet_field}">${facet_field}</div>
<script type="text/javascript">
  facet_array = new Array();
  #foreach($facet in $facets)
    facet_array.push(['${facet.key}', ${facet.value}])
  #end
  var chart = new JSChart('jschart_${chart_type}_${facet_field}', '${chart_type}');
  chart.setDataArray(facet_array);
  chart.setTitle('${facet_field}')
  chart.draw();
</script>
```

http://localhost:8983/solr/select?q=*&rows=0&facet=on&facet.field=content_type&wt=velocity&v.template=jschart&v.layout=layout&jschart.type=pie&title=Pie

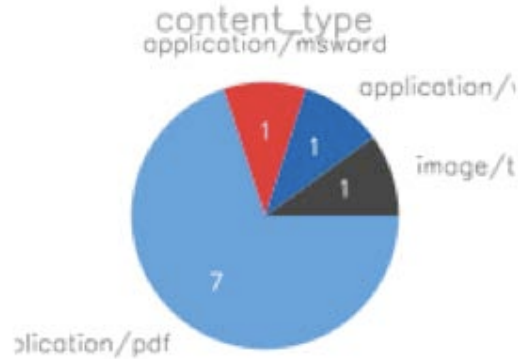
Cleaning up chart URLs

added to solrconfig.xml

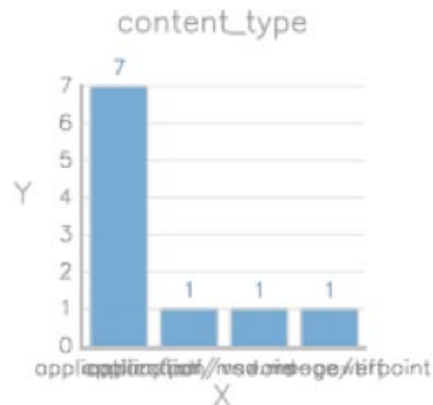
```
<requestHandler name="/jschart" class="solr.SearchHandler">
  <lst name="defaults">
    <!-- UI settings -->
    <str name="wt">velocity</str>
    <str name="v.template">jschart</str>
    <str name="jschart.type">pie</str>
    <!-- results details -->
    <str name="rows">0</str>
    <!-- query parsing -->
    <str name="defType">lucene</str>
    <str name="q">*:*</str>
    <!-- faceting -->
    <str name="facet">on</str>
    <str name="facet.field">content_type</str>
    <str name="facet.mincount">1</str>
  </lst>
</requestHandler>
```

Standalone views

<http://localhost:8983/solr/jschart?v.layout=layout&jschart.type=pie>



<http://localhost:8983/solr/jschart?v.layout=layout&jschart.type=bar>



Ajaxifying

added to browse.vm, inside facet field loop

```
<a href="#"
onClick="javascript:$('#jschart_${field.name}'
).load('/solr/jschart?jschart.type=pie&q=${esc.url($params.get('q'))}');">Pie</a>
```

```
<a href="#"
onClick="javascript:$('#jschart_${field.name}'
).load('/solr/jschart?jschart.type=bar&q=${esc.url($params.get('q'))}');">Bar</a>
<div id="jschart_${field.name}"></div>
```

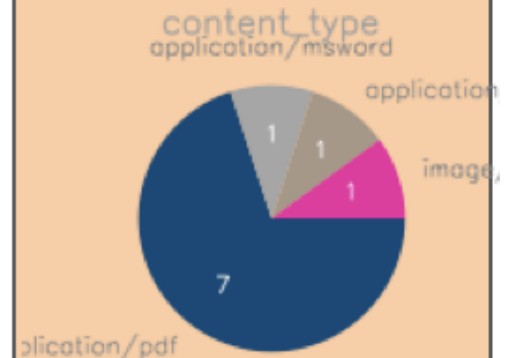
jQuery is included in the default layout

Facets

content_type

- [application/pdf](#) (7)
- [application/msword](#) (1)
- [application/vnd.ms-powerpoint](#) (1)
- [image/tiff](#) (1)

Pie Bar



`debugQuery=true`

- ▶ Adds scoring explanations for each hit
- ▶ dumps the request and response objects (`toString`) at the bottom of the page

Score Explanation

http://localhost:8983/solr/browse?q=user+interfaces&debugQuery=true

[/Users/erikhatcher/dev/lucid/presentations/webinars/prototyping-june-2010/docs/edUi09-SolrFlair.pdf](#)

Last modified: Fri Jun 04 11:08:44 PDT 2010

... Solr Flair Search *User Interfaces* Powered by Apache Solr Erik Hatcher erik ... choice for an extensible *user interface* development framework Blacklight @ UVa Blacklight ... soon: lucenebook.com LucidFind <http://www.lucidimagination.com/search/?q=user+interface> Food ...

[toggle explain](#)

```
0.13626574 = (MATCH) sum of:
  0.059439037 = (MATCH) weight(text:user in 6), product of:
    0.59465766 = queryWeight(text:user), product of:
      0.90468985 = idf(docFreq=10, maxDocs=10)
      0.65730554 = queryNorm
    0.09995505 = (MATCH) fieldWeight(text:user in 6), product of:
      2.828427 = tf(termFreq(text:user)=8)
      0.90468985 = idf(docFreq=10, maxDocs=10)
      0.0390625 = fieldNorm(field=text, doc=6)
  0.07682671 = (MATCH) weight(text:interfac in 6), product of:
    0.80397904 = queryWeight(text:interfac), product of:
      1.2231436 = idf(docFreq=7, maxDocs=10)
      0.65730554 = queryNorm
    0.09555809 = (MATCH) fieldWeight(text:interfac in 6), product of:
      2.0 = tf(termFreq(text:interfac)=4)
      1.2231436 = idf(docFreq=7, maxDocs=10)
      0.0390625 = fieldNorm(field=text, doc=6)
```

Now what?

- **Script the indexer**
- **Customize header & footer, adjust styles and colors, add your logo**
- **Show your boss**
- **Ask "what now?"**

General next steps

- **Script full & incremental indexing processes**
- **Adjust schema**
 - ▼ fields, field types, analysis
- **Tweak configuration as needed**
 - ▼ caches, indexing parameters
- **Deploy to staging/production environments**

Is it done?

No.

Keep it (slightly) ugly, for this reason.

▼ iron out capabilities, then pretty it up

prototyping provides the Solr requests your REAL application will use. Copy and paste what you need from Solr's logs and prototype templates

Prototyping tools

- **CSV update handler**
- **Schema Browser (in Solr's admin)**
- **Solritas**
- **Solr Explorer**
 - ▼ <https://issues.apache.org/jira/browse/SOLR-1163>
- **Solr Flare**
 - ▼ <http://wiki.apache.org/solr/Flare>

- **Performance**
- **Scalability**
- **Relevance**
- **Automate all of the above,
start baselines and avoid regressions**

Questions?

Thank You!

