

SAN FRANCISCO MAY 25-26 2011

The Once and Future History of Enterprise Search and Open Source

Marc Krellenstein, CTO
marc@lucidimagination.com



LUCENE
REVOLUTION

Presented by

lucid
IMAGINATION

Apache
Solr 

Lucene

Evolving challenges in full text search

- Finding something in a lot of content (recall, scalability)
 - **IBM/STAIRS vs. US gov/Basis, BRS, Dialog, Verity**
 - **Lycos and Fast , Infoseek, Excite → AltaVista**
 - **Centralized search → Distributed search (Fast, Google, Lucene/Solr)**
- Finding just the good stuff (precision)
 - **SMART, Autonomy, Google, Lucene/Solr, authority scores, browsing/clustering/faceting,...**
- Finding it fast (performance)
 - **Fast, Google, Lucene/Solr**
- Making it easy (simplicity)
 - **Google, Lucene/Solr***
- Deploying good search everywhere (all of the above plus price, flexibility)
 - **Lucene/Solr**

Google

- Breakthrough in precision of Internet search
 - **Popularity algorithm hides the bad stuff**
 - **Proved importance of understanding data & users**
 - **Set expectations for accuracy of enterprise search**
- Set a new standard for search performance
 - **Sub-second (or near)**
- Proved value of good adaptive spell-checking
- Demonstrated the power of distributed search for scale
- Reinforced the importance of simplicity and a single search box
- Proved the value of search
 - **Search needs to be everywhere**

But Google is not like most enterprise search applications



- Google
 - Most data is bad, many good enough answers...task is to screen out the bad
 - Many privacy issues among users
 - No security issues
 - Many naïve users with little patience...speed is important
- Enterprise search
 - Most or all the data may be good, often only one answer to a search need
 - Many security issues
 - Few or no privacy issues between users
 - Naïve and sophisticated users motivated by an organizational purpose
- The best enterprise search tools will fit enterprise needs

Best practice recall and precision



- Recall
 - **Percent of relevant documents (items) returned**
 - **50 good answers in system, 25 returned = 50% recall**
- Precision
 - **Percent of documents returned that are relevant**
 - **100 returned, 25 are relevant = 25% precision**
- Ideal is 100% recall and 100% precision: return all relevant documents and only those
- 100% recall is easy – return all documents...but precision so low they can't be found...precision harder
- Need adequate recall & enough precision for the task
 - **That will vary by application (data & users)...**

How to get good recall

- Collect, index and search all the data
 - **Check for missing or corrupt data**
 - **Index everything – stop words not usually needed today**
 - **Search everything...limit results by category AFTER the search (clustering/faceting)**
- Normalize the data
 - **Convert to lower case, strip/handle special characters, stemming, ...**
- Use spell-checking, synonyms to match users' vocabulary with content
 - **Adaptive spell-checking, application-specific synonyms**
- Light (or real) natural language processing for abstract concepts
 - **'Recent documents on Asia'**

How to get good precision

- Term frequency (TF) – more occurrences of query terms is better
- Inverse document frequency (IDF) – rarer query terms are more important
- Phrase boost – query terms near each other is better
- Field boost – where the query term is in doc matters (e.g., in ‘title’ better)
- Length normalization – avoid penalizing short docs
- Recency – all things being equal, recent is better
- Authority – items linked to, clicked on or bought by others may be better
- Implicit and explicit relevance feedback, more-like-this – expand query (queries usually underdetermined...intent??)
- Clustering/faceting – when above fail or intent is not specific
- Lots of data...Watson, Google Translate

The emergence of open source Lucene/Solr



- Lucene
 - **Built in late 90's by Doug Cutting.... Apache release 2001**
 - **State of the art Java library for indexing and ranking... many ports since**
 - **Contributed to open source to keep it going and reusable**
 - **Wide acceptance by 2005, mostly by technology organizations, products**
- Solr
 - **Build in 2005 by Yonik Seeley to meet CNET needs for quicker-to-build applications and faceting...had to be open source...Apache release 2006**
 - **Lucene over HTTP, schema, cache management, replication,... and faceting**
- Open source as a development model, not a religion
- 4,000+ sites – Apple, Cisco, EMC, HP, IBM, LinkedIn, MySpace, Netflix, Salesforce, Twitter, Gov, Wikipedia...

Current Lucene/Solr: strengths

- Best practice segmented index (like Google, Fast)
 - **Scalability via SolrCloud distributed search → billions of documents**
- Best practice, flexible ranking (term/field/doc boosts, function queries, custom scoring...)
- Best overall query performance and complete query capabilities (unlimited Boolean operations, wildcards, find-similar, synonyms, spell-check...)
- Multilingual, query filters, geo search, memory mapped indexes, near real-time search, advanced proximity operators...
- Rapid innovation
- Extensible architecture, complete control (open source)
- No license fees (open source)
- **CORE TECHNOLOGY AS GOOD OR BETTER THAN ANY OTHER...AND OPEN SOURCE**

Open source Lucene/Solr: weaknesses



- Those typical of open source
 - **No formal support**
 - **Limited access to training, consulting**
 - **Lack of stringent integrated QA**
 - **Pace of development and open source environment too complex for some (e.g., what version should I download? What patches? GUI?)**
- Others
 - **Lucene/Solr development has tended to focus on core capabilities, so missing certain features for enterprise search (e.g., connectors, security, alerts, advanced query operations)**

Addressing open source Lucene/ Solr weaknesses



- Lucene/Solr Community
 - **Apache Lucene/Solr community has a wealth of information on web sites, wikis and mailing lists**
 - **Community members usually respond quickly to questions**
- Consultants
 - **May be especially helpful for systems integration or addressing gaps**
- Commercialization
 - **Companies commercializing open source provide commercial support, certified versions, training and consulting...may fill in gaps or address ease of use**
 - **Examples: Red Hat, MySQL ,Lucid Imagination**
- Internal resources – usually in combination with one or more of the above

Product strengths of top commercial competitors



- Well established players tend to be full-featured
- Some organizations have focused on a particular application or domain (e.g., ecommerce, publishing, legal, help desk)
- Some competitors have focused on appliance-like simplicity

Weaknesses of top commercial competitors

- Usually expensive, especially at scale
- Platform or portability limitations
- Limited transparency
- Limited flexibility, especially for other than intended application or domain
- Limited customization, especially for appliance-like products
- Sometimes limited scalability
- Technical debt and/or lack of rapid innovation
- Customers are dependent on the company's continued business success

Current competitive landscape



- For last 5 years commercial companies have felt increasing competition from Lucene/Solr because of the combination of its capability and price
 - **Very hard to justify multi-million dollar deals given Lucene/Solr**
 - **Lucene/Solr sometimes wins on performance alone**
- Some competitors have responded with diversification
 - **Re-invent themselves as a business intelligence or other kind of company**
 - **Produce search derivative applications**
 - **Focus on specific domains**
- Some have been acquired
- But the need for good, affordable, flexible search remains

The competitive future

- Basic search has become commoditized and widespread...but
 - **Top commercial companies usually often have one or more key weaknesses**
 - **Existing search is often mediocre and too expensive or difficult to maintain, grow or customize/enhance**
 - **Producing best practice search is still hard (and search remains a hard problem...intent, context, NLP...)**
- Market strength and features of competitors will keep competitors going a while...but
 - **Very hard to justify high prices, especially for large applications**
 - **Very hard to justify closed and proprietary technology**
- Lucene/Solr capabilities, performance, control, price and continued rapid innovation (and addressing weaknesses) will likely lead to its dominance

Resources

- *Lucene in Action, Second Edition*, by Michael McCandless, Erik Hatcher and Otis Gospodnetic. Manning, 2010.
- *Solr 1.4 Enterprise Search Server*, by David Smiley and Eric Pugh. Packt Publishing, 2009.
- Solr reference guide:
<http://www.lucidimagination.com/Downloads/LucidWorks-for-Solr/Reference-Guide>