

Agenda

1. User Interface [Kevin]
2. Information Retrieval Process [Pritom]
3. Indexing & Searching [Petros]
4. Ranking Methods [Kevin]
5. Performance & Example Queries [Oliver]
6. Challenges [all]
7. Future Improvements [Oliver]

Information Retrieval Process

1. Title [String]
2. Date [Date]
3. Views [Integer]
4. Tags [String[]]
5. Related Question [String[]]
6. Question
 1. Question [String]
 2. Code [String]
 3. Votes [Integer]
7. Answer
 1. Answer [String]
 2. Accepted Answer [Boolean]

stack overflow NEW Search...

Home PUBLIC Stack Overflow Tags Users Jobs Teams Q&A for work Learn More

Comparison of full text search engine - Lucene, Sphinx, Postgresql, MySQL?

asked 10 years ago
viewed 133,360 times
active 4 years, 3 months ago

Featured on Meta
Announcing the arrival of Valued Associate #879: Cesar Manara

Linked

- 0 full text search in java web application
- 192 Choosing a stand-alone full-text search server: Sphinx or SOLR?
- 63 Why is Solr so much faster than Postgres?
- 28 Full Text Searching with Rails
- 22 How to find similarity between mySQL rows?
- 14 php mysql fulltext search: lucene, sphinx, or?
- 5 Any way to reduce the size of texts?
- 4 mySQL LIKE query
- 13 Full text search(Postgres) Vs Elastic search
- 5 Google like search query using Full Text Search in SQL Server

see more linked questions...

Related

- 14 php mysql fulltext search: lucene, sphinx, or?
- 192 Choosing a stand-alone full-text search server: Sphinx or SOLR?
- 45 How to evaluate hosted full text search solutions?
- 414 ElasticSearch, Sphinx, Lucene, Solr, Xapian. Which fits for which usage?
- 3 MySQL full-text search in Rails?
- 2 MySQL Full-Text Search Across Multiple Tables - Quick/Long Solution?
- 264 NoSQL (MongoDB) vs Lucene (or Solr) as your database
- 54 Any reason not use PostgreSQL's built-in full text search on Heroku?
- 0 Moving from Full text search in Sql Server To MySql
- 1 Full text search in PostgreSQL JSON (schema-less, NoSQL-like)

Hot Network Questions

- How can I support myself financially as a 17 year old with a loan?
- Why are prions in animal diets not destroyed by the digestive system?

I'm building a Django site and I am looking for a search engine.

A few candidates:

- Lucene/Lucene with Compass/Solr
- Sphinx
- Postgresql built-in full text search
- MySQL built-in full text search

Selection criteria:

- result relevance and ranking
- searching and indexing speed
- ease of use and ease of integration with Django
- resource requirements - site will be hosted on a [VPS](#), so ideally the search engine wouldn't require a lot of RAM and CPU
- scalability
- extra features such as "did you mean?", related searches, etc

Anyone who has had experience with the search engines above, or other engines not in the list -- I would love to hear your opinions.

EDIT: As for indexing needs, as users keep entering data into the site, those data would need to be indexed continuously. It doesn't have to be real time, but ideally new data would show up in index with no more than 15 - 30 minutes delay

mysql postgresql full-text-search lucene sphinx

share improve this question edited Dec 16 '11 at 20:53 asked Apr 10 '09 at 10:38

ajreal 41.8k • 10 • 73 • 112 Continuation 5,361 • 15 • 64 • 96

25 26: MySQL fulltext search and transactions are (presently) mutually exclusive. MySQL fulltext indexes require the MyISAM table type, which doesn't support transactions. (As opposed to the InnoDB table type which supports transactions, but not fulltext indexes.) - Carl G Jan 31 '10 at 20:56

1 PostgreSQL full-text search, [Tsearch](#) does not support phrase search. However, it's on the TODO list [salmsu.ru/~megera/wiki/FTS_Todo](#) - Gnanam Dec 9 '10 at 14:13

1 Anyone looking at this for Django should checkout the haystack app. [haystacksearch.org](#) - Keyo Jul 2 '11 at 4:56

4 [slideshare.net/billkanwin/...](#) - Acute May 7 '12 at 6:40

23 @CarlG, Just for everybody's reference. MySQL 5.6+ has Full text search support with innodb engine - DhruvPathak Dec 18 '12 at 13:36

add a comment

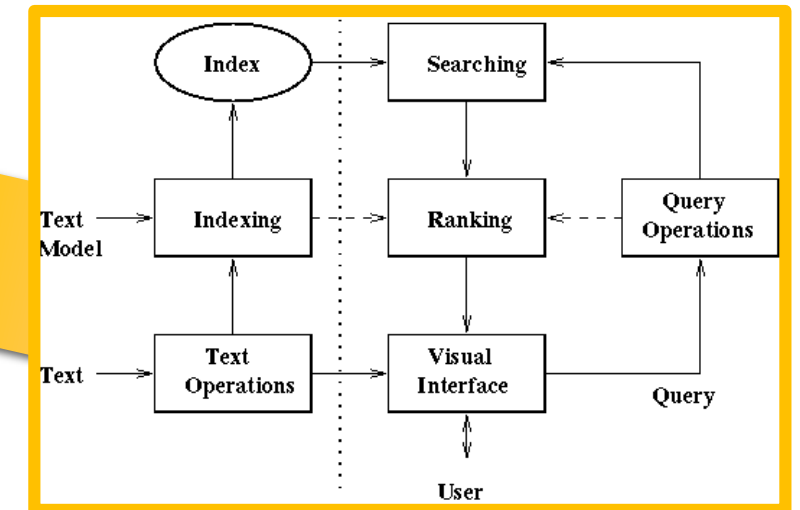
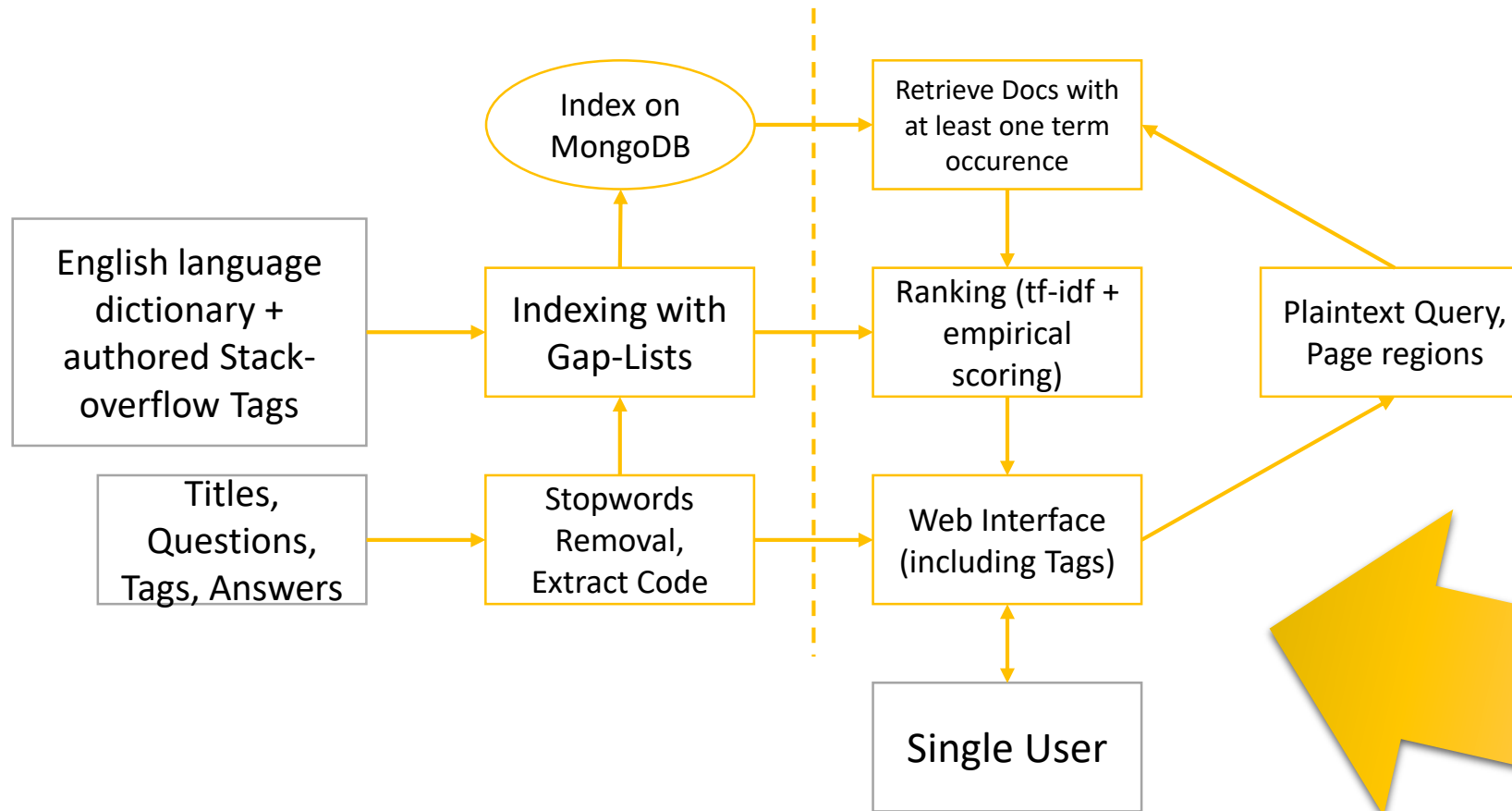
8 Answers active oldest votes

Good to see someone's chimed in about Lucene - because I've no idea about that.

Sphinx, on the other hand, I know quite well, so let's see if I can be of some help.

- Result relevance ranking is the default. You can set up your own sorting should you wish, and give specific fields higher weightings.
- Indexing speed is super-fast, because it talks directly to the database. Any slowness will come from complex SQL queries and un-indexed foreign keys and other such problems. I've never noticed any slowness in searching either.
- I'm a Rails guy, so I've no idea how easy it is to implement with Django. There is a Python API that comes with the Sphinx source though.
- The search service daemon (searchd) is pretty low on memory usage - and you can set limits on [how much memory](#) the indexer process uses too.
- Scalability is where my knowledge is more sketchy - but it's easy enough to copy index files to multiple machines and run several searchd daemons. The general impression I get from others

Information Retrieval Process



Performance & Example Queries

1. Think of developer's information need and corresponding queries
2. Use Google Search for only Stackoverflow results (one day for calibration)
3. Filter for date range 06/05/2017 to 06/05/2019
4. Cross-validate results with Stackoverflow built-in Elasticsearch results
5. Return ranked top 5 questions

User query text	nginx reverse-proxy
Regional Search	-
Result 1	https://stackoverflow.com/questions/54151/nginx-reverse-proxy
Result 2	https://stackoverflow.com/questions/54038
Result 3	https://stackoverflow.com/questions/54123/for-nextcloud-not-work
Comments	

User query text	java ide
Regional Search	-
Result 1	https://stackoverflow.com/questions/51803/docker-nginx-php7-fpm-and-xdebug