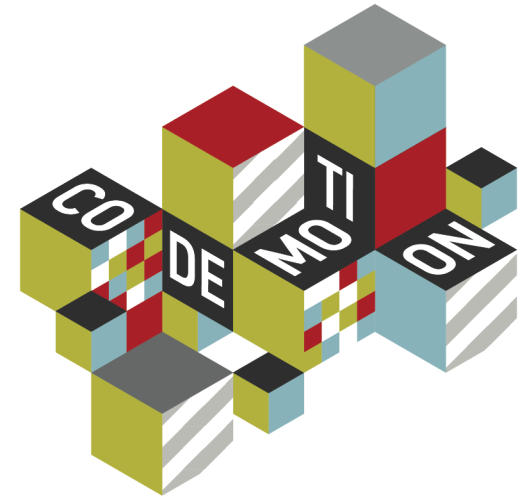


# Sphinx: Open Source Search Server

---

Matteo Baccan

matteo@baccan.it  
Jug Novara, PUG Novara



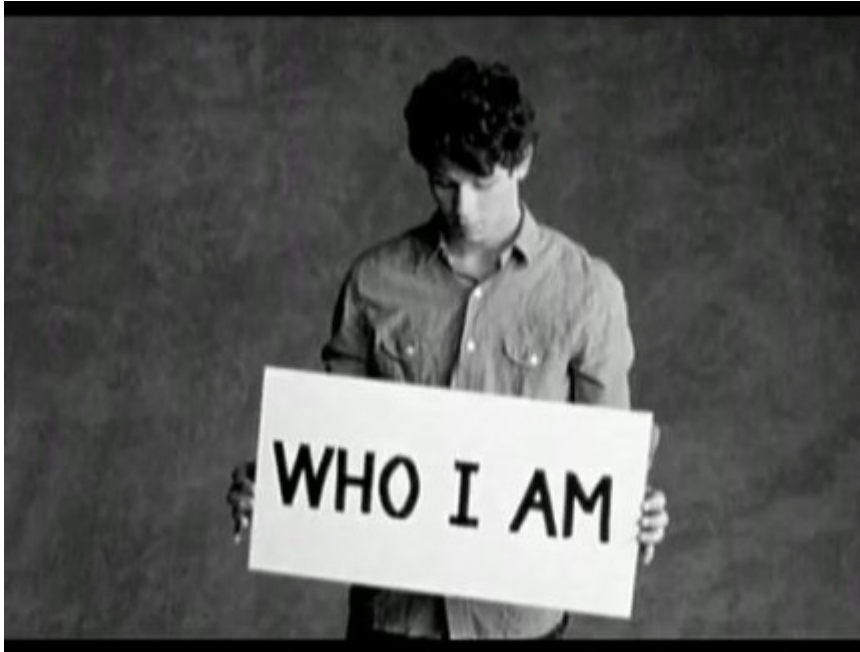
# Sphinx: Open Source Search Server

- Relatore: **Matteo Baccan**
- Area: Open Source
- Target: Expert, Developer, Power User
- Difficoltà: Media-Bassa

Sporchiamoci le mani con un'alternativa a Lucene e MySQL. Un prodotto OpenSource, portabile, veloce, attento alle prestazioni e ai consumi di spazio e memoria. Vedremo come distribuire il carico, come effettuare hotswap di indici, come usare lo stesso JDBC per accedere a MySQL e Sphinx, come gestire indici dinamici e statici. Utilizzeremo query, subquery, strategia di ricerca diverse. Studieremo le migliori tecniche per indicizzare milioni di documenti, il tutto tramite PHP e JAVA su casi reali

Da 0 a 100 in 40 minuti, allacciate le cinture ;)

# Chi sono?



Divulgatore informatico, giornalista per riviste tecniche nazionali oltre 700 articoli per :Dev, Login, Computer Programming, Mokabyte. Relatore tecnico ad eventi di programmazione: Borland Forum 2000, Webbit 2004, JIP day 2005, Javaday (2006,2007,2010) PHPDay (2008, 2010), CONFSL 2010, WebTech 2010, Codemotin (2011,2012). Autore di Corso di C# ISBN 8881500167. Autore di JobCrawler e HTML2POP3 (Oltre 900.000 download su SourceForge)

# Cos'è Sphinx?

- Acronimo di **SQL Phrase Index**
- Un motore di ricerca “full text”
- Un “emulatore” MySQL
- Veloce, Compatto, Scalabile

# Perché usare Sphinx?

- Uso già MySQL/Lucene/Solr no
- Uso Lucene in deep no
- Uso Lucene, ma è lento a indicizzare forse
- Ho molto spazio allocato dagli indici forse
- Con java non conosco alternative forse
- Devo convertire molto codice SQL si
- Non ho un motore di ricerca si
- Ho bisogno di una soluzione veloce si

## Alcuni numeri?

- 10-1000x volte più veloce di MySQL sulle ricerche full-text
  - MySQL è veloce solo con gli indici in RAM)
- 2-3x volte più veloce sulle ricerche non full-text
  - Sphinx ha meno feature da supportare ;)
- 2-4x volte più veloce di Lucene sulle ricerche full-text  
[fonte Sphinx Technologies Inc.] ;)

## Risultati rilevanti

- Sphinx ha diversi algoritmi per il calcolo del rank
  - phrase proximity (aka LCS) : posizione delle parole
  - BM25 : frequenza delle parole

La combinazione di questi algoritmi può variare la qualità dei dati cercati

## Alcuni algoritmi di rilevanza

- SPH\_RANK\_PROXIMITY\_BM25: combina LCS e BM25
- SPH\_RANK\_BM25: solo BM25
- SPH\_RANK\_NONE: nessun rank
- SPH\_RANK\_WORDCOUNT: solo per numero di parole, in base alla priorità dei campi
- SPH\_RANK\_SPH04: LCS e BM25, ma portando in testa gli exact match
- SPH\_RANK\_EXPR: basato su espressione



## Ma chi lo usa?

- boardreader.com, 1.000.000 query/day, 16.000.000.000 documenti, 5TB, 27 sphinx box
- craigslist.org, 200.000.000 query/day (2.000 al secondo) 15 sphinx box

<http://sphinxsearch.com/info/powered/>

- moonlab.it network: 500.000 query/day, 10.000.000 documenti, 300 millesimi tempo medio di risposta 2 sphinx box bilanciate da macchine dual core con 4g di RAM (con jboss, mysql, sphinx e altri processi in concorrenza)

## Architettura base

- sphinx.conf file di configurazione
- Indexer indicizzatore
- SearchD demone di ricerca
- Search ricerca command line

## Come indicizzo?

- Definisco un datasource : connessione e stringa di selezione dei dati
- Definisco un indice: posizione nel file system e tipologia
- Indicizzo: dico all'indexer di creare l'indice appena definito

# Come indicizzo? Il datasource

```
source job {  
  type           = mysql  
  sql_host       = localhost  
  sql_user       = sphinx  
  sql_pass       =  
  sql_db         = jobcrawler  
  sql_port       = 3306  
  sql_query      = SELECT id, descrizione FROM `job`  
  sql_query_info = SELECT id, titolo FROM job WHERE id=$id  
}
```

# Come indicizzo? L'indice

```
index job {  
    source          = job  
    path            = ./data/job  
    docinfo         = none  
    mlock           = 0  
    morphology      = none  
    min_word_len    = 1  
    charset_type    = sbcs  
    html_strip      = 0  
}
```

# Come indicizzo? L'indexer

## **indexer.exe job**

Sphinx 2.0.4-id64-release (r3135)

Copyright (c) 2001-2012, Andrew Aksyonoff

Copyright (c) 2008-2012, Sphinx Technologies Inc (<http://sphinxsearch.com>)

using config file './sphinx.conf'...

indexing index 'job'...

collected 125565 docs, 61.9 MB

sorted 8.9 Mhits, 100.0% done

total 125565 docs, 61894430 bytes

**total 7.319 sec, 8456618 bytes/sec, 17155.91 docs/sec**

total 2 reads, 0.155 sec, 12707.9 kb/call avg, 77.6 msec/call avg

total 67 writes, 0.258 sec, 890.2 kb/call avg, 3.8 msec/call avg

## Come cerco?

- Da riga comando con Search
- Lanciando il demone SearchD e interrogandolo con
  - SphinxAPI (PHP, Java e altri)
  - SphinxQL : tramite SQL
  - SphinxSE : MySQL storage engine

# SphinxAPI - Java

```
SphinxClient cl = new SphinxClient( "localhost", 3307 );
SphinxResult res = cl.Query("j2ee php", "job");
if ( res!=null ) {
    for ( int i=0; i<res.matches.length; i++ ) {
        SphinxMatch info = res.matches[i];
        System.out.println( (i+1) + ". id=" + info.docId );
    }
}
```



# SphinxAPI - PHP

```
require ( "sphinxapi.php" );  
$cl = new SphinxClient ();  
$cl->SetServer( "127.0.0.1", 3307 ); $cl->SetArrayResult ( true );  
$res = $cl->Query( "j2ee php", "job" );  
if ( $res ) {  
    $n = 0;  
    foreach ( $res["matches"] as $docinfo ) {  
        $n++; print "$n. doc_id=$docinfo[id]<br/>";  
    }  
}
```

# SphinxQL

Sphinx è in grado di emulare il protocollo MySQL  
Al suo interno è presente un interprete SQL

**MySQL non è più necessario**

Molti client sono in grado di usare Sphinx come se  
fosse un server MySQL

## SphinxQL - JDBC

```
Connection conn =  
DriverManager.getConnection("jdbc:mysql://127.0.0.1:3308");  
Statement stmt = conn.createStatement();  
ResultSet rs = stmt.executeQuery("select * from job  
where match('j2ee php')");  
while( rs.next() )  
    System.out.println( rs.getString(1) );
```

# SphinxSE

- Occorre ricompilare MySQL
  - Oppure scaricarlo da <http://www.coreseek.cn>

- Si verifica guardando gli engine installati

SHOW ENGINES

	Engine	Support	Comment
<input type="checkbox"/>	MyISAM	DEFAULT	Default engine as of MySQL 3.23
<input type="checkbox"/>	MEMORY	YES	Hash based, stored in memory, u
<input type="checkbox"/>	InnoDB	YES	Supports transactions, row-leve
<input type="checkbox"/>	BerkeleyDB	YES	Supports transactions and page-
<input type="checkbox"/>	BLACKHOLE	YES	/dev/null storage engine (anyth
<input type="checkbox"/>	EXAMPLE	YES	Example storage engine
<input type="checkbox"/>	ARCHIVE	YES	Archive storage engine
<input type="checkbox"/>	CSV	NO	CSV storage engine
<input type="checkbox"/>	SPHINX	YES	Sphinx storage engine 1.11-dev
<input type="checkbox"/>	ndbcluster	NO	Clustered, fault-tolerant, memo
<input type="checkbox"/>	FEDERATED	YES	Federated MySQL storage engine
<input type="checkbox"/>	MRG_MYISAM	YES	Collection of identical MyISAM
<input type="checkbox"/>	ISAM	NO	Obsolete storage engine

# SphinxSE

```
CREATE TABLE jobsp (  
    id          BIGINT,  
    weight      INTEGER NOT NULL,  
    query       VARCHAR(3072) NOT NULL,  
    group_id    INTEGER,  
    INDEX(query)  
) ENGINE=SPHINX  
CONNECTION="sphinx://localhost:3307/job";
```

# SphinxSE

La tabella si usa mettendola in Join con la tabella contenente i dati, indicando come query il dato da cercare

```
SELECT job.id, titolo FROM job
```

```
JOIN jobsp ON (job.id=jobsp.id)
```

```
WHERE QUERY='j2ee php;mode=any';
```

## Indici distribuiti

- In caso di alto traffico, è possibile creare indici dati dalla somma dei risultati di differenti istanze di Sphinx e indici locali
- Quello che varia, rispetto all'indice standard, è la dichiarazione di agent di ricerca, al posto/in concomitanza di indici fisici

# Indici distribuiti – nodo master

```
index job {  
    type          = distributed  
  
    agent          = localhost:3310:job  
    agent          = localhost:3311:job  
  
    agent_connect_timeout = 1000  
    agent_query_timeout   = 3000  
}
```



## Indici distribuiti – interrogazione nodi

```
java -jar sphinxapi.jar -p 3310 -i job j2ee php
```

'j2ee' found 1005 times in 632 documents

'php' found 337 times in 255 documents

```
java -jar sphinxapi.jar -p 3311 -i job j2ee php
```

'j2ee' found 933 times in 598 documents

'php' found 371 times in 268 documents

# Indici distribuiti – interrogazione master

```
java -jar sphinxapi.jar -p 3312 -i job j2ee php
```

'j2ee' found 1938 times in 1230 documents

'php' found 708 times in 523 documents

## RT Index (RealTime)

- Indici dinamici in grado di gestire
  - INSERT
  - UPDATE
  - DELETE
  - SELECT
- Più dinamici, ma tendono a degradare le performance per frammentazione e su grandi quantità di dati

## RT Index

```
index rt {  
    type          = rt  
    path          = data/rt  
    rt_field      = descrizione # full text  
    rt_attr_uint  = gid  
}
```

# RT Index

## Non è MySQL è Sphinx

```
Class.forName("org.gjt.mm.mysql.Driver");  
Connection conn =  
DriverManager.getConnection("jdbc:mysql://127.0.0.1:3320");  
Statement stmt = conn.createStatement();  
    stmt.execute("delete from rt where id in(1,2,3)");  
    stmt.execute("insert into rt values(1,'primo record' ,100)");  
    stmt.execute("insert into rt values(2,'secondo record',111)");  
    stmt.execute("insert into rt values(3,'terzo record' ,122)");
```

# RT Index

```
mysql -h 127.0.0.1 -P 3320
```

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 1

Server version: 2.0.4-id64-release (r3135)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

```
mysql>
```

# RT Index

```
mysql> select * from rt;
```

id	weight	gid
1	1	100
2	1	111
3	1	122

```
3 rows in set (0.00 sec)
```

```
mysql> select * from rt where match('secondo');
```

id	weight	gid
2	1680	111

```
1 row in set (0.00 sec)
```

# Di cosa non abbiamo parlato

- XMLPipe datasource
- Main e Delta index
- Index merge
- Stopwords
- Wordforms
- Sorting Mode
- Group by
- .... e molte altre cose ...





## Qualche link

- <http://www.sphinxsearch.com/> Sphinx (source, info)
- <http://www.coreseek.cn/news/14/99/> SphinxSE (binari)
- <http://www.moonlab.it> Cosa facciamo
- <http://www.baccan.it> Cosa faccio



Matteo Baccan  
[matteo@baccan.it](mailto:matteo@baccan.it)