

# Full Text Search Intro



By : Adi Baron, CTO Tikal

# Agenda

---

- What is search?
- Where's the need?
- Concepts
- Search in relational databases
- Full text search
- What's next?



# What is search?

- Search is vague



# What is search?

- Search is also mandatory (!)

The screenshot displays the AllMusic website interface. At the top, there's a navigation bar with links for 'Account', 'Wishlists', and 'Home'. A search bar is prominently featured with a 'GO' button. Below the navigation bar, a large red banner advertises 'FREE SHIPPING'. The main content area is divided into several sections: a 'I graduated in' section with a grid of state abbreviations, a 'classmates.com' link, and a search bar for 'Artist/Group'. To the right, there's a 'site menu' with links like 'New Releases', 'Music Videos', and 'Top Searches'. Below the search bar, there's a section for 'album of the day' featuring Vivaldi's 'Cello Concertos'. To the left, there's a section titled 'TURN YOUR PC INTO A MUSIC DISCOVERY MACHINE' with a 'DOWNLOAD zune 3.0' button. At the bottom, there's a section for 'AllMusic's Favorite Albums of 2008' and 'THE ALLMUSIC BLOG' with a post about 'AllMusic's Favorite Rock Albums of 2008'. The right sidebar contains 'new releases' and 'The AllMusic Gift Guide'.



# What is search?

---

- Integrating efficient search into your application is not an easy task.



# Where's the need?

---

- Modern information systems give users access to more and more information
- A user that knows precisely **what** to find and **where** to look is an edge case
  - (in this situation there's actually no need for a search)
- Most of the time **what** and **where** are blurry



# Where's the need?

---

- Before knowing where to look, we need to have a decent understanding of what we are looking for.  
(Surprisingly, some users barely know what they are looking for...)
- Users seek
  - Help
  - Guidance
  - Search refinement
  - Relevance



# Where's the need?

---

- Even when the users know precisely what they are looking for, they might not know how to access the information
- Information systems are expected to provide access to data:
  - Efficiently
  - Fast
  - Without pieces of irrelevant information





# Concepts

---

- Refining the **what** can be addressed by:
  - Categorizing information
  - Using a detailed search screen
  - Using a “simple” search box



# Categorizing Information

The image displays three overlapping web pages, illustrating categorization of information. The background page is ThinkGeek, featuring a navigation menu with categories like T-Shirts & Apparel, Geek Toys, Gadgets, Home & Office, Computing, Caffeine, Electronics, Geek Kids, and Books. It also has a 'FREE SHIPPING' banner and a 'New Stuff!' section. The middle page is AllMusic, showing a search bar, a 'FREE SHIPPING' banner, and a 'New Stuff!' section. The foreground page is The AllMusic Blog, featuring a 'TURN YOUR PC INTO A MUSIC DISCOVERY MACHINE' advertisement, a 'FREE SHIPPING' banner, and a 'New Stuff!' section.

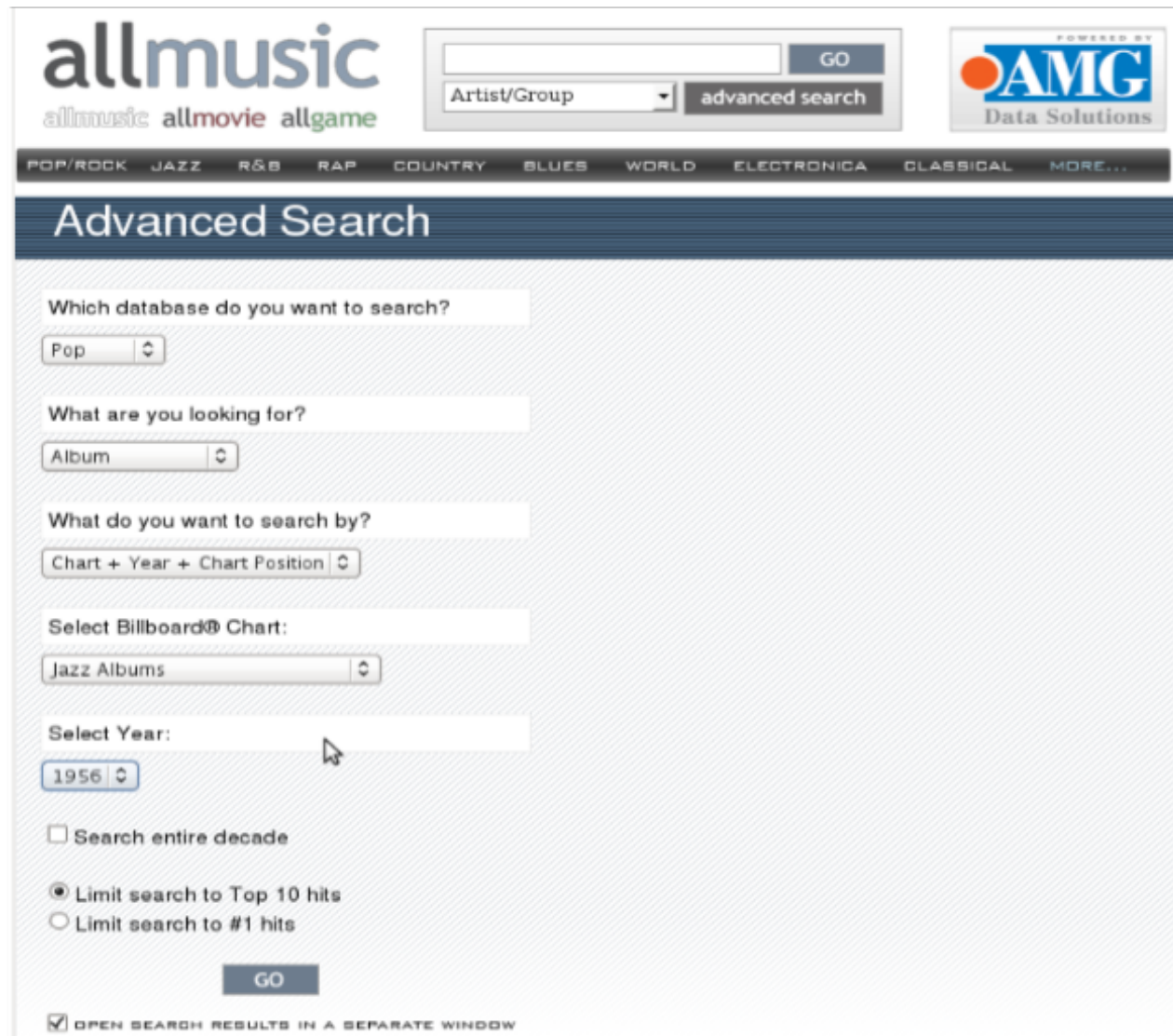

# Categorizing Information

---

- Done by business experts during data insertion.
- Cons:
  - Might not match the mindset of the users
  - Almost impossible when there is too much information
- Pros:
  - Beneficial if the user has no predefined idea
  - A system for navigating the application



# Detailed search screen



**allmusic**  
allmusic allmovie allgame

GO  
Artist/Group advanced search

POWERED BY  
**AMG**  
Data Solutions

POP/ROCK JAZZ R&B RAP COUNTRY BLUES WORLD ELECTRONICA CLASSICAL MORE...

## Advanced Search

Which database do you want to search?  
Pop

What are you looking for?  
Album

What do you want to search by?  
Chart + Year + Chart Position

Select Billboard® Chart:  
Jazz Albums

Select Year:  
1956

☐ Search entire decade

☒ Limit search to Top 10 hits  
☐ Limit search to #1 hits

GO

☒ OPEN SEARCH RESULTS IN A SEPARATE WINDOW

# Detailed search screen

---

- Cons:

- Not friendly to beginner users

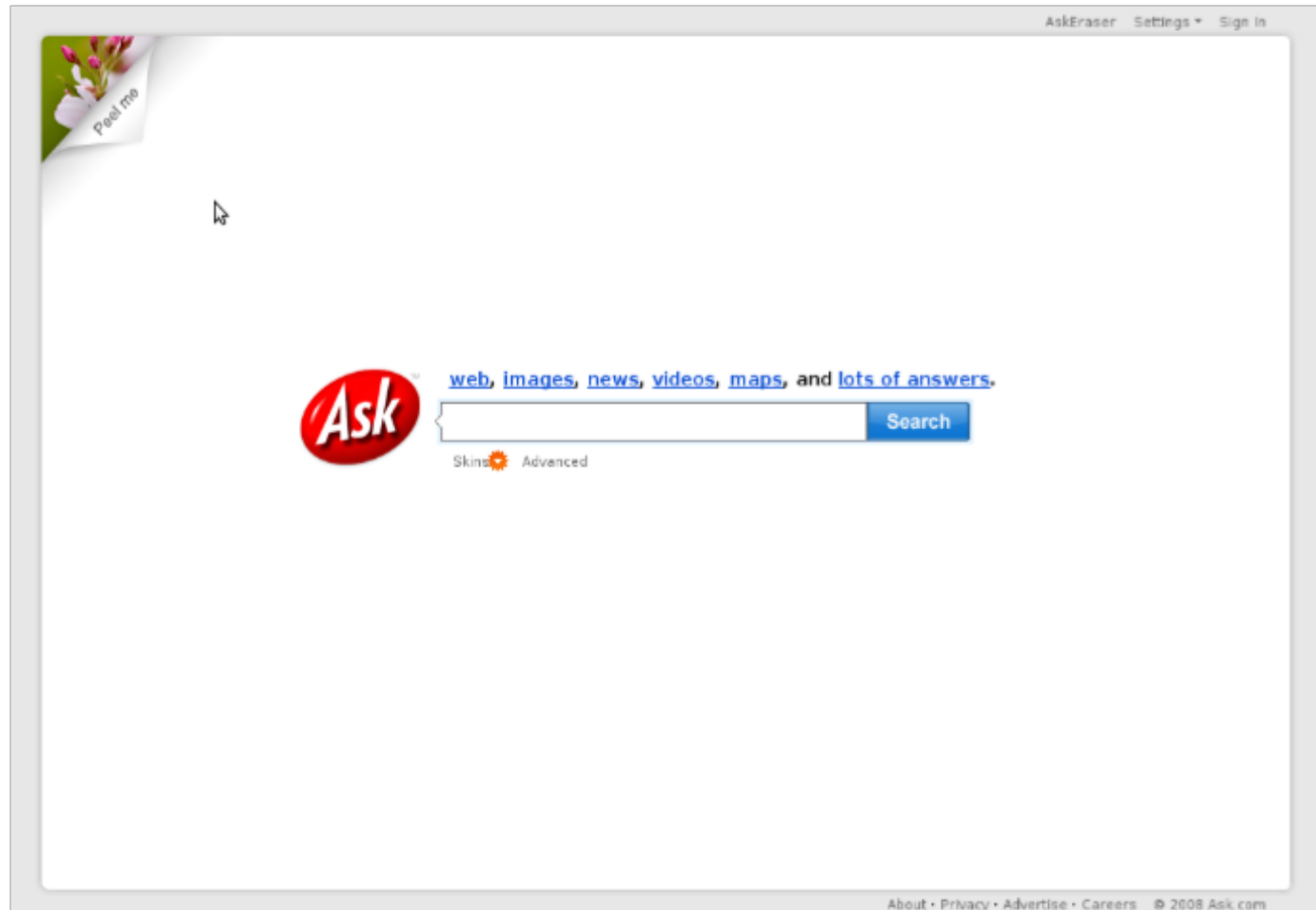
- Pros:

- Very efficient when the user knows what to look for
  - Expert users can fine-tune their search
- Gives an idea of what to search for, in a “simple” search box
- Easier to implement





# A “simple” search box



# A “simple” search box

---

- Cons:
  - Hard to implement
    - Make the user use the language of the system
- Pros:
  - Best user experience for beginners (when properly implemented)



# Search in relational DB

---

- RDBMS and SQL are great tools.
- Can they answer our search needs?
- A quick example:



# Search in relational DB

---

- A user is looking for a CD on riaa-fans.com
  - riaa-fans.com use SQL to answer search queries
  - Users are provided with simple search box
- The user query
  - “a collection cd of jez music”



# Search in relational DB

---

- The CD table:
  - `ID` bigint(20) NOT NULL auto\_increment,
  - `TITLE` varchar(50) NOT NULL,
  - `DESCRIPTION` varchar(250) NOT NULL,
  - `GENRE` varchar(50) NOT NULL,
- A possible query:

```
select cd.id
from CD cd
where cd.TITLE = ? OR cd.GENRE = ? OR cd.DESCRPTION like ? OR...
```



# Search in relational DB

---

- We're searching **words**, not **columns**
- What about spelling mistakes (“jez”)?
- Search words from the same family
- Relevance...



# Full text search

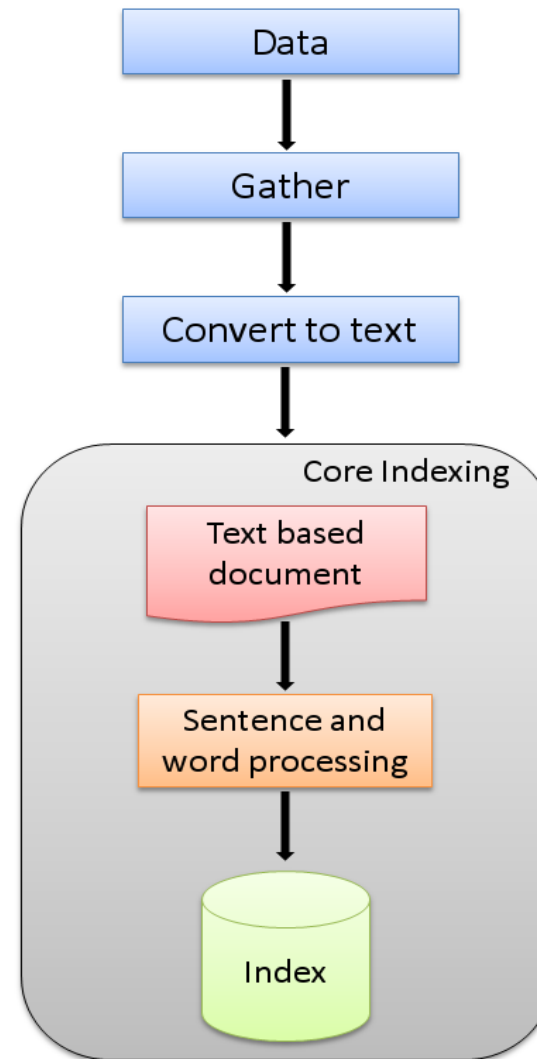
---

- A promising solution
- How?
  - Indexing
  - Searching



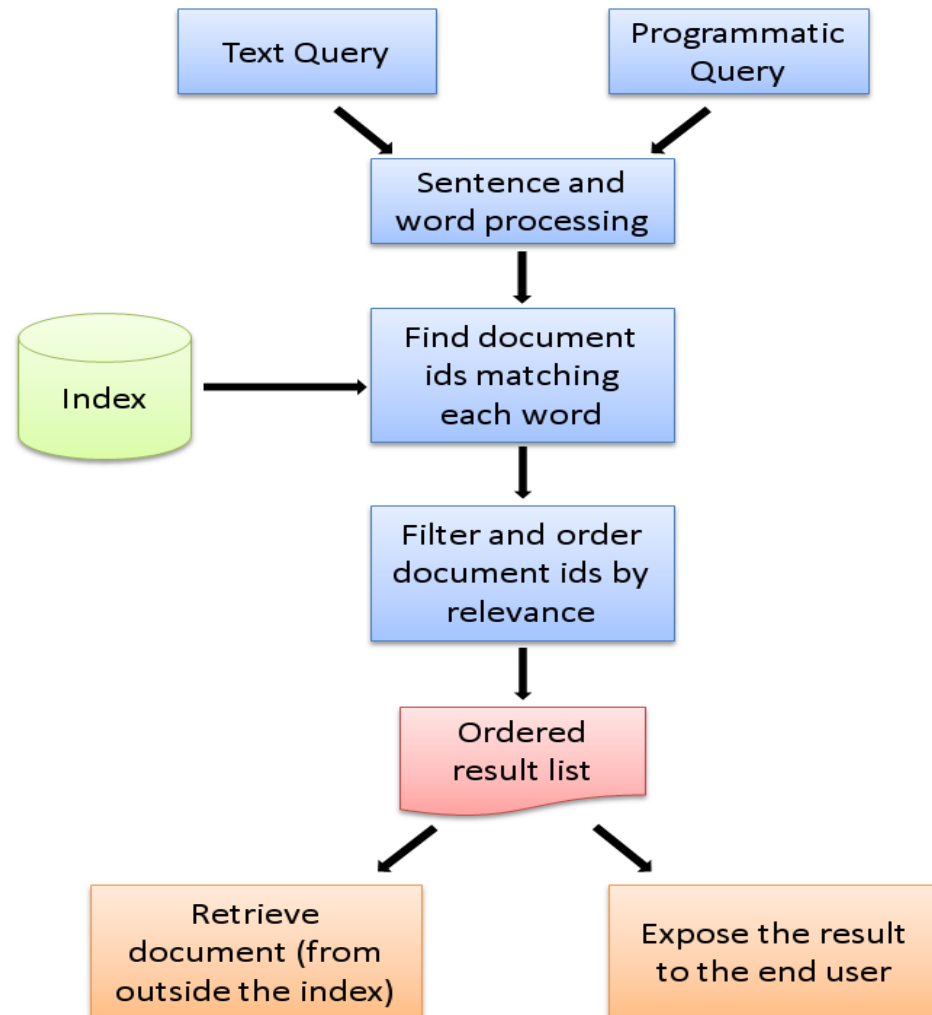
# Full text search

- Indexing



# Full text search

- Searching



# Full text search

---

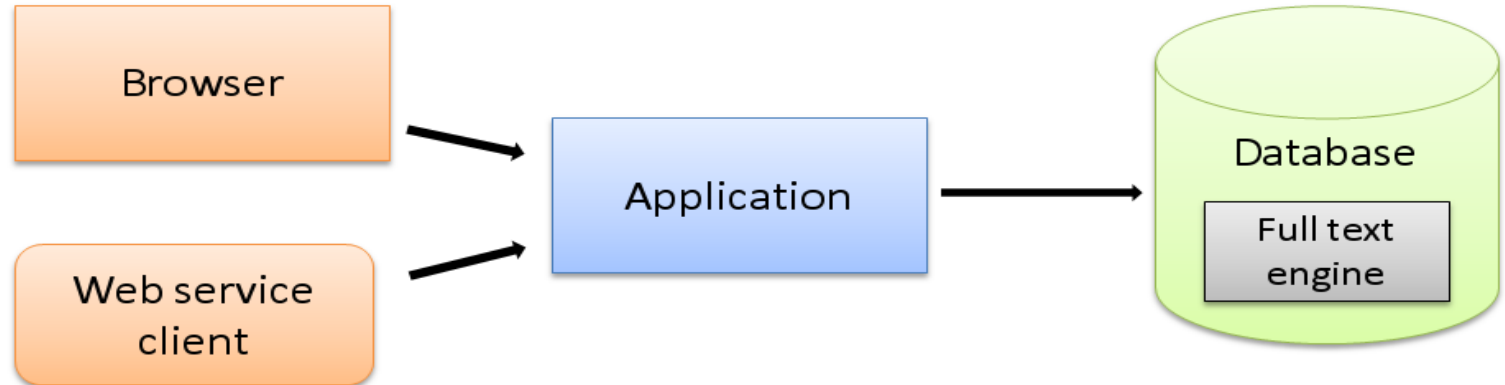
- Solutions
  - An integrated full-text search in the relational database engine
  - A black box server providing the full-text service
  - A library providing a full-text engine implementation





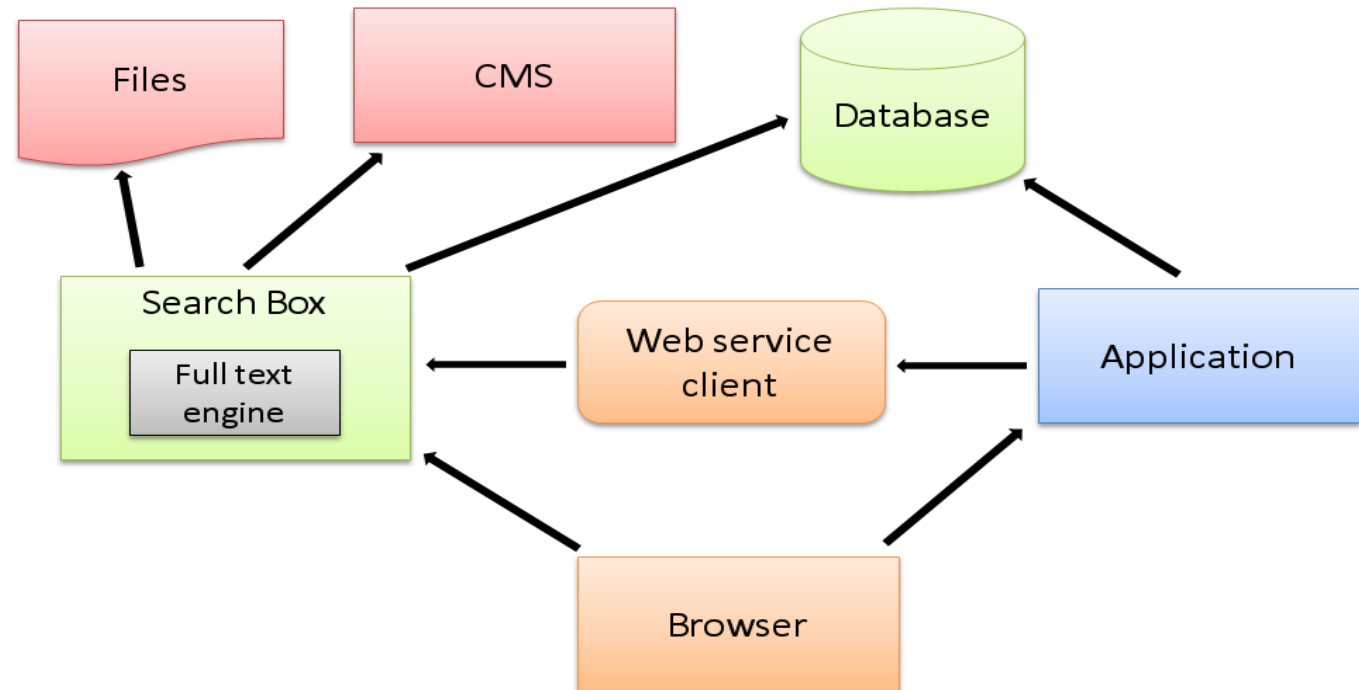
# Full text search

- An integrated full-text search in the relational database engine



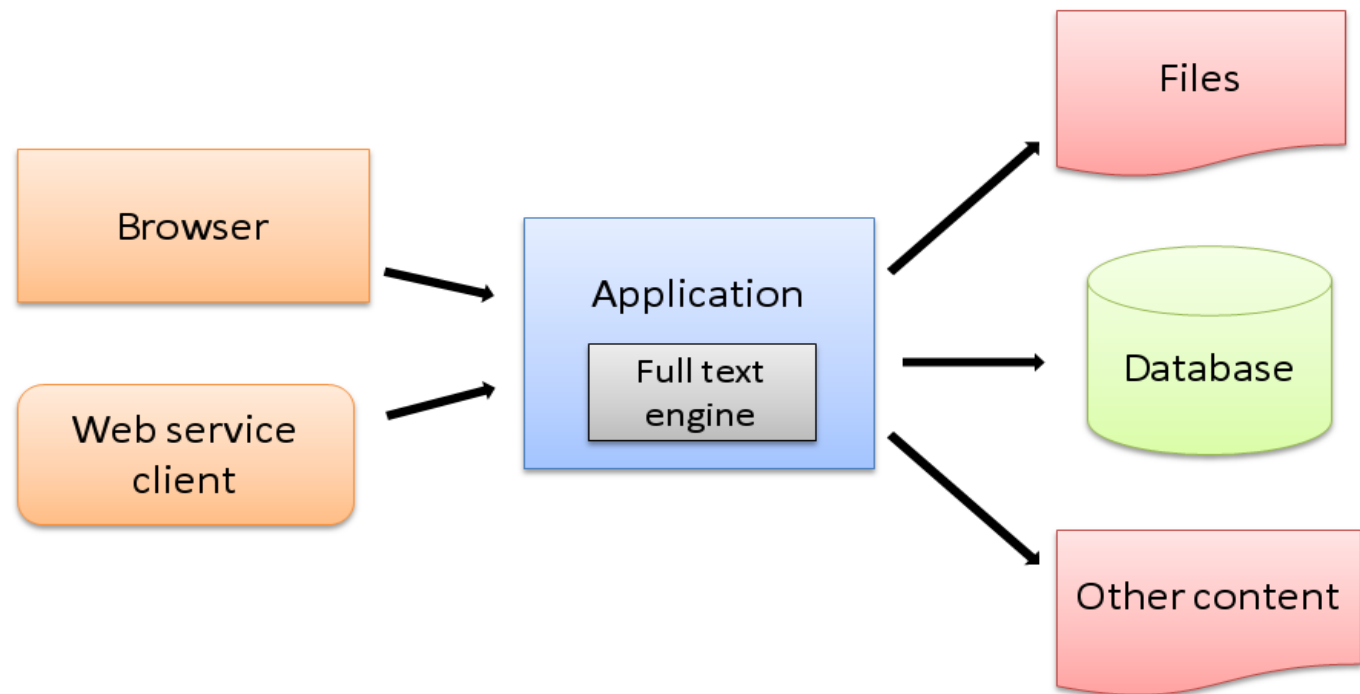
# Full text search

- A black box server providing the full-text service



# Full text search

- A library providing a full-text engine implementation



# What's next?

---

- Hibernate Search and Compass
  - Use a Library
    - Lucene



# What's next?

---

- Lucene

- Open-Source library (Java)
- Library - Not Application
- Simple API
- +George -eat, Apple -pie +Tiger, animal:monkey AND food:banana







# Q&A