Search Engine

Big Data

Goal

Develop a search engine using an inverted index structure

Inverted index is a type of indexing used by search engines and document-oriented databases such as Google, MongoDB, Elastic Search or Apache Solr

It allows quick searching of text documents

Inverted index

For each word, it saves the documents that contain the word

When a user enters a specific search term, it is very fast to know the documents that contain that term.

There are two types:

- Record-level: each word contains a list of references to documents
- Word-level: additionally contains the positions of each word within a document and the frequency of the word.

Structure examples

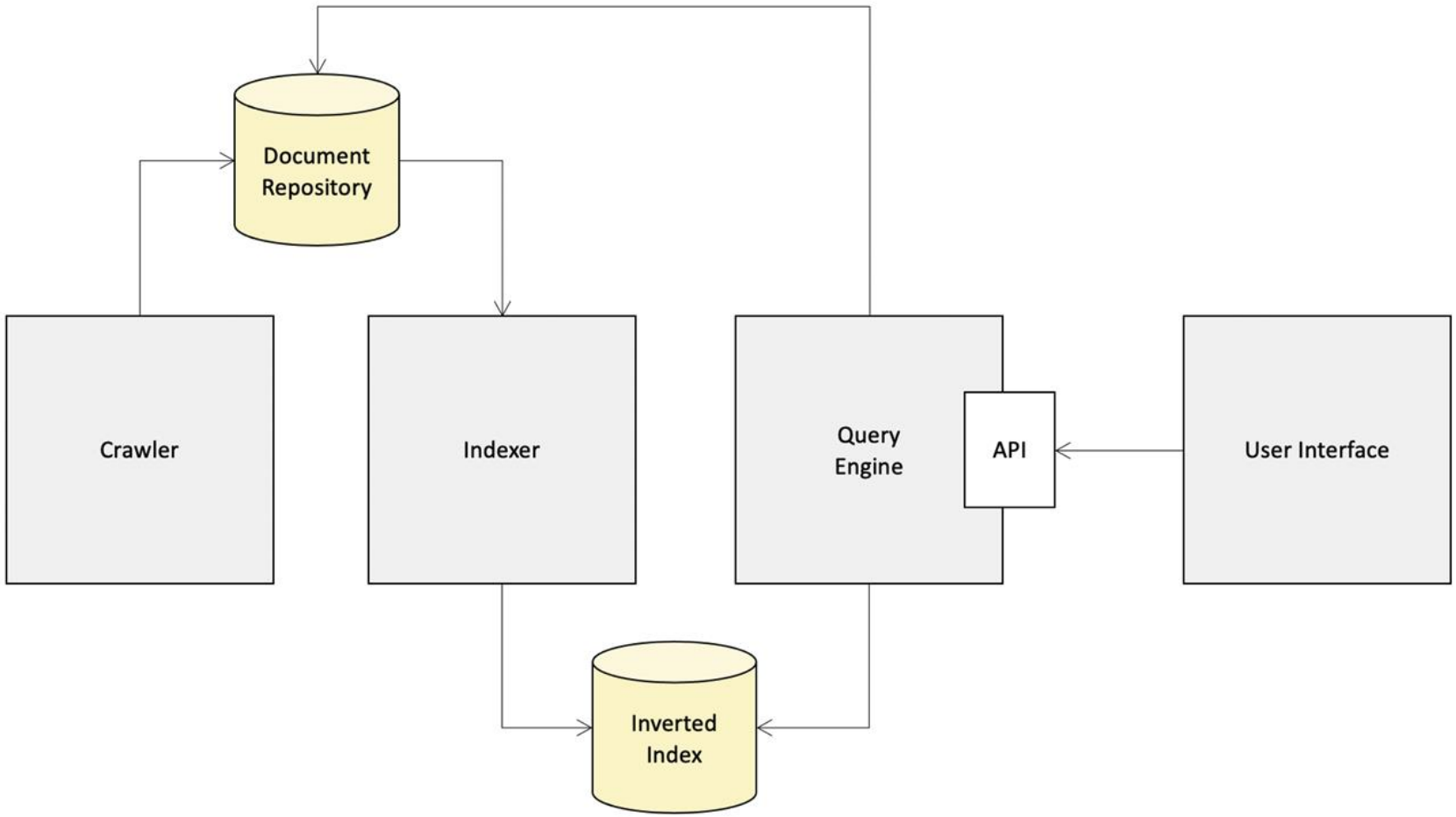
Original documents:

- ("001", "the car is nice")
- ("002", "that car is mine")
- ("003", "the car is the best")

Inverted Index

- ("best", ["003"])
- ("car", ["001", "002", "003"])
- ("the", ["001", "003"])
- ...

Solution architecture



Crawler

Download documents periodically

Books are stored in the document repository

Books will be retrieved from:

https://www.gutenberg.org/

Indexer

Index documents periodically when repository is updated

Each document is processed to feed

- Metadata database (authors, year, language…)
- Inverted index (documents and positions where the word appears)

Stop words must be avoided

Query engine

Based on query terms, searches documents in

- inverted index and,
- metadata store

It provides a REST API that should follow a specifc signature

User interface (optional)

Client that uses the search engine API

Provides support to enter terms for search in

- Inverted index
- Metadata

Shows the results of querying the datamarts