
Open WebSearch
The Inverted File in Search
Gijs Hendriksen
OpenWebSearch.EU Webinar
March 10th, 2025
Open Web Search
SUPPORTED BY
NGI
Funded by the European Union

Document ranking

➜ Given a query, compute relevance scores for each document in the index

➜ Higher score: document is more likely to be relevant

➜ Different ranking models possible
- Term-based (tf-idf, BM25)
- Feature-based (learning-to-rank)
- LLMs

Tf-idf document ranking

- Term frequency (tf): how often does a query term occur in the document?

  -> How relevant is the document to that query term?

- Inverse document frequency (idf): how many documents contain the query term?

  -> How "interesting" is the term?

Document ranking

Web search engines crawl the web …

The Open Web Search project …

The open-source software …

Document ranking

Web search engines crawl the web ...

Score = 0.45

The Open Web Search project ...

Score = 0.41

The open-source software ...

Score = 0

Creating an Inverted File (indexing)

Web search engines crawl the web ...

The Open Web Search project ...

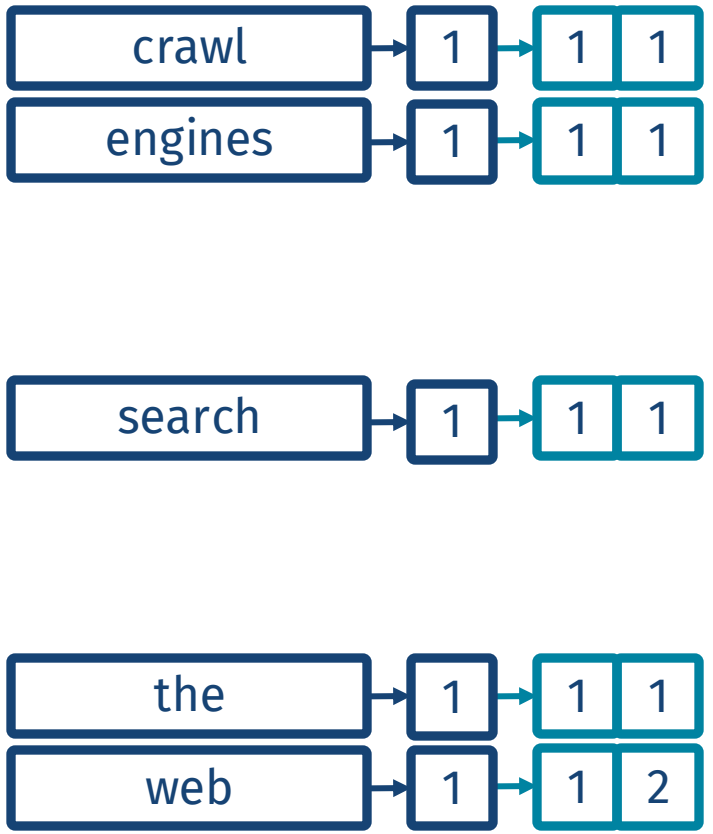
The open-source software ...

Creating an Inverted File (indexing)

Web search engines crawl the web …

The Open Web Search project …

The open-source software …

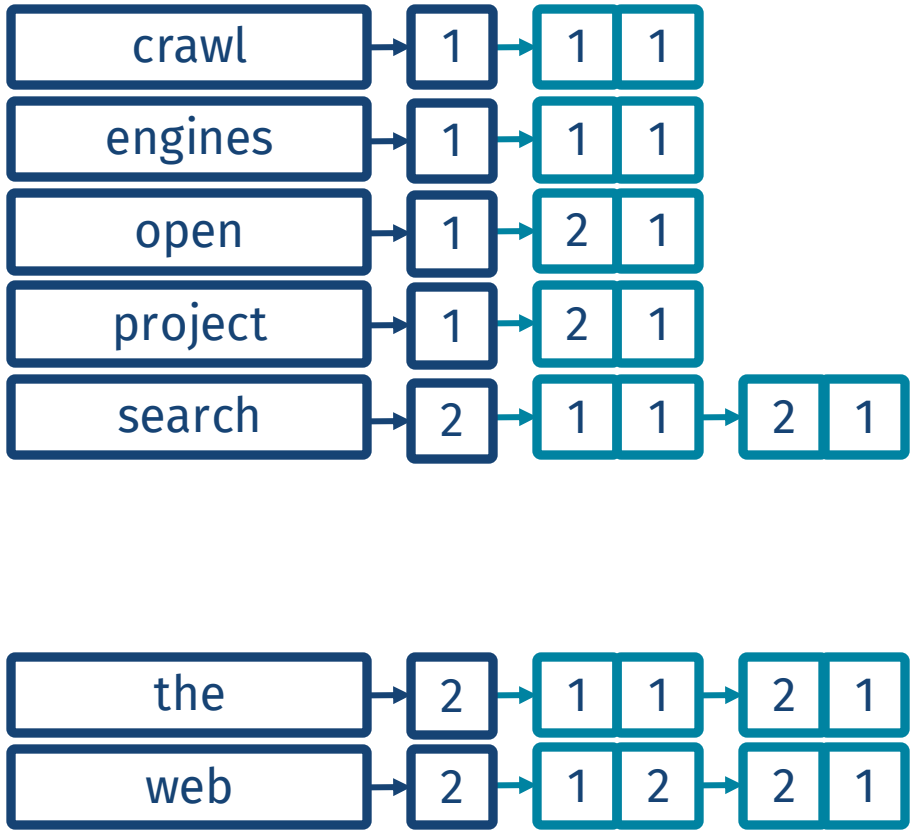


Creating an Inverted File (indexing)

Web search engines crawl the web …

The Open Web Search project …

The open-source software …

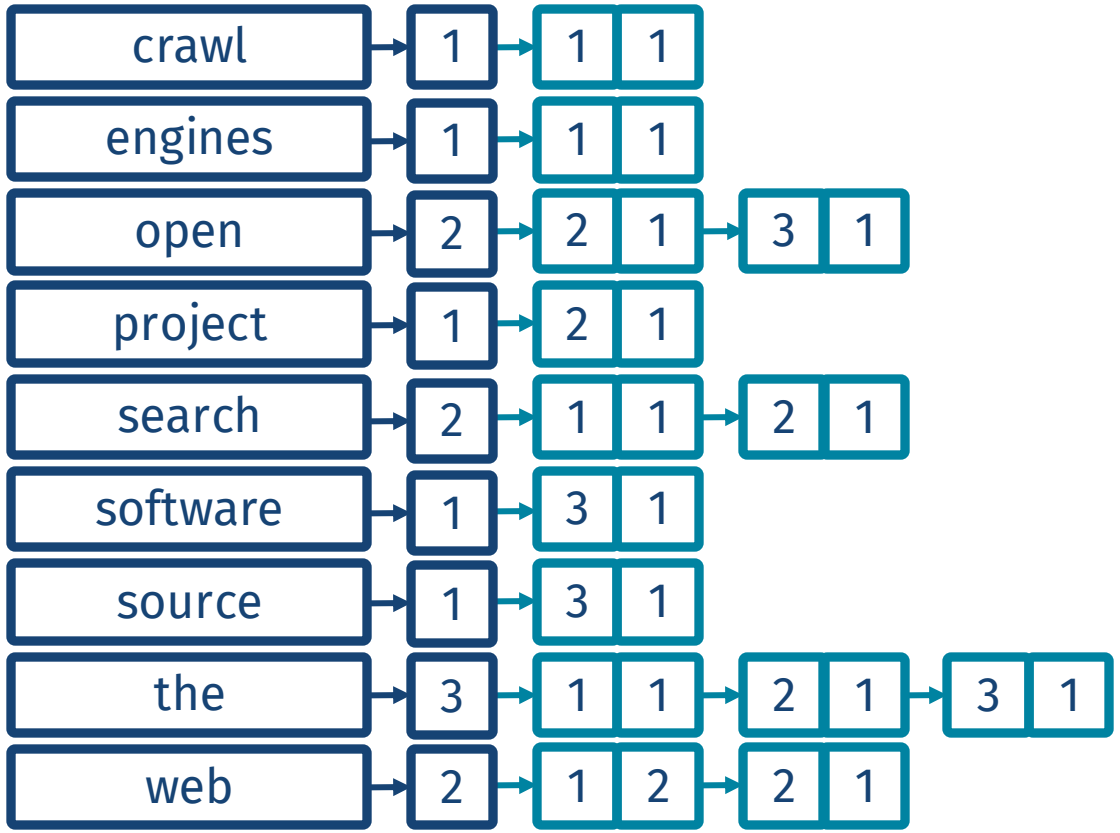


Creating an Inverted File (indexing)

Web search engines crawl the web …

The Open Web Search project …

The open-source software …



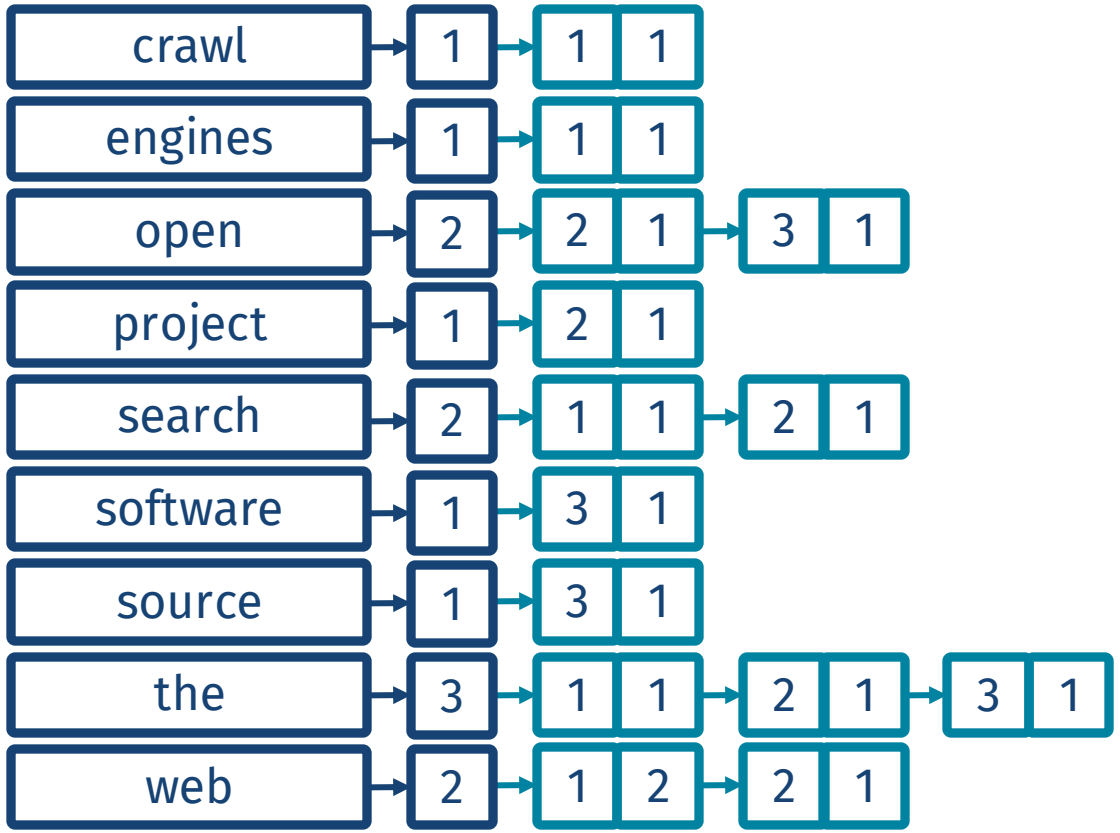
Using an Inverted File (retrieval)

web search

Web search engines crawl the web …

The Open Web Search project …

The open-source software …



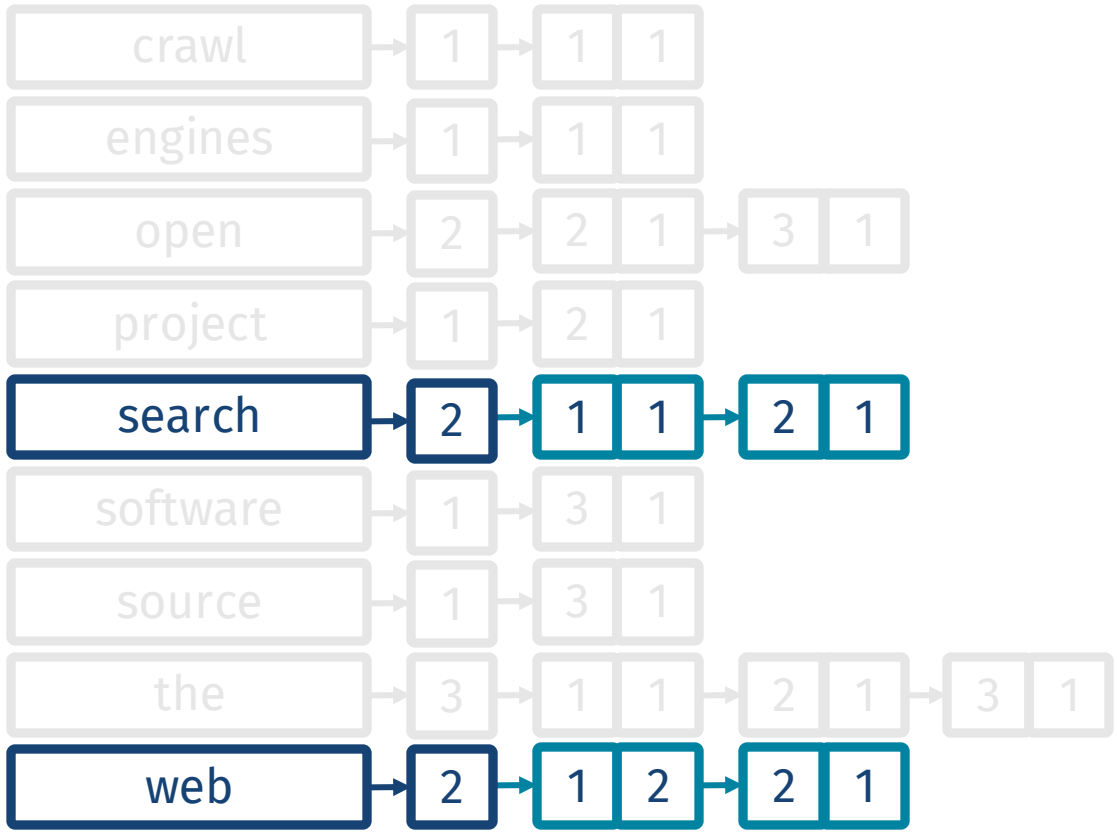
Using an Inverted File (retrieval)

web search

Web search engines crawl the web …

The Open Web Search project …

The open-source software …



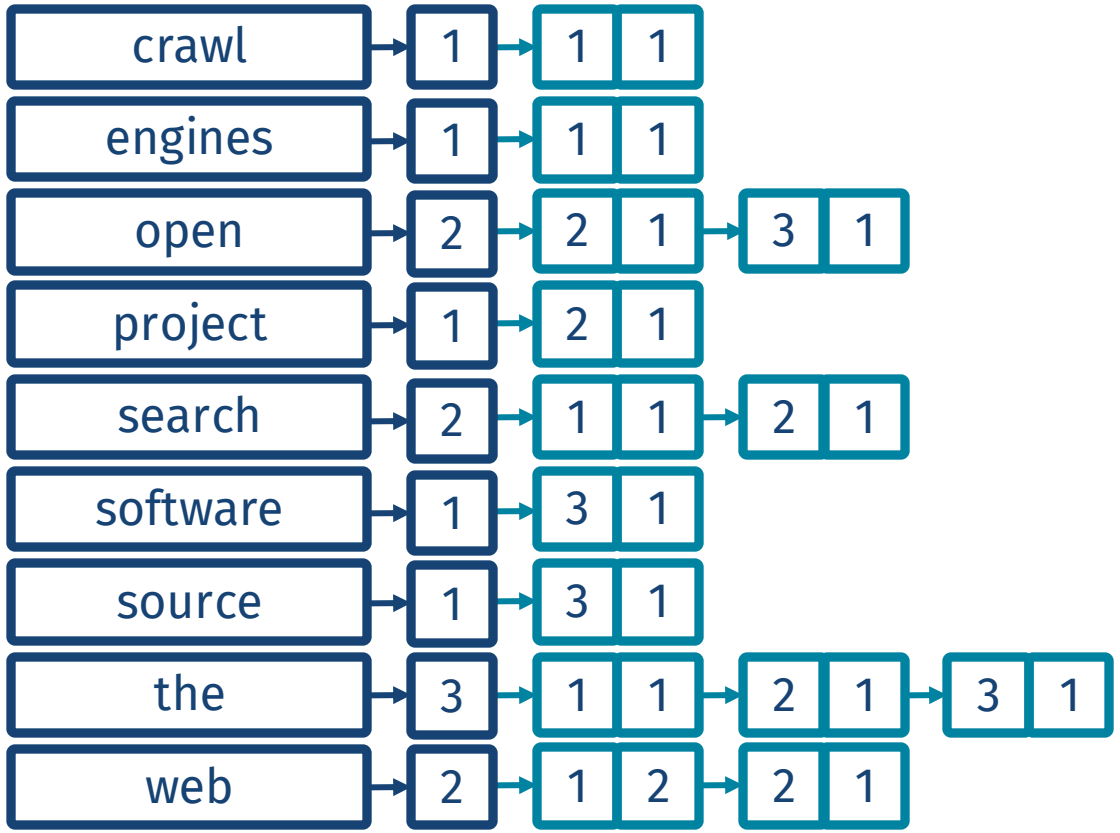
Stopwords

the web

Web search engines crawl the web …

The Open Web Search project …

The open-source software …

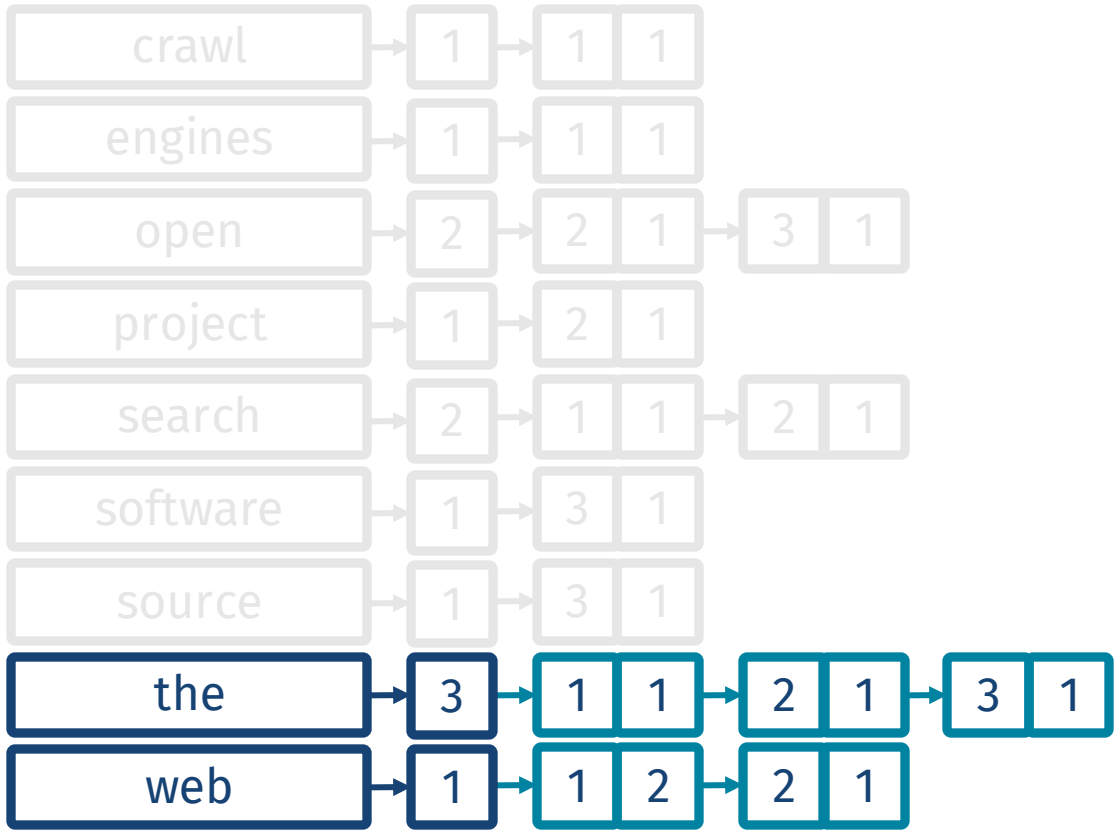


Stopwords

Web search engines crawl the web ...

The Open Web Search project ...

The open-source software ...

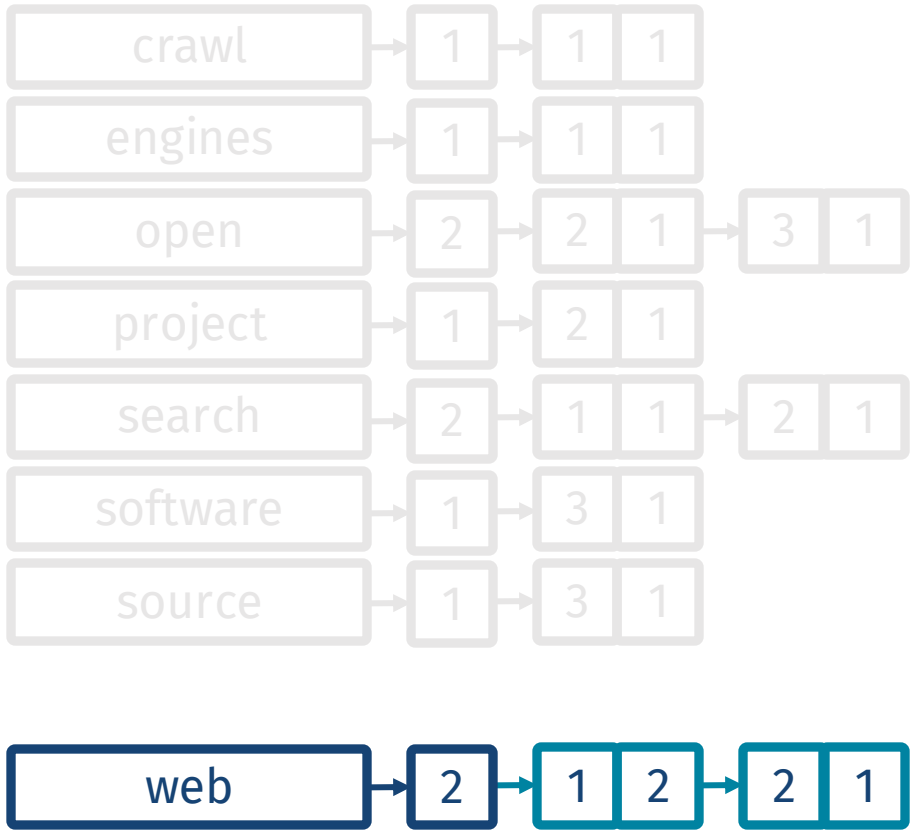


Stopwords

Web search engines crawl the web ...

The Open Web Search project ...

The open-source software ...



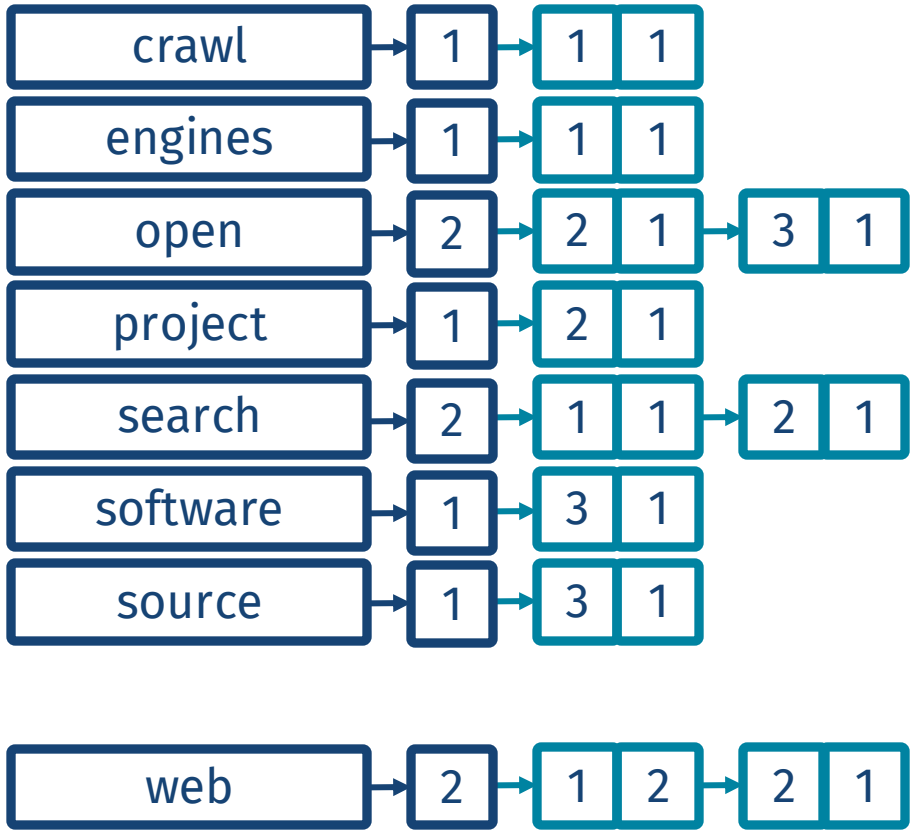
Stemming

crawling engine

Web search engines crawl the web …

The Open Web Search project …

The open-source software …



Stemming

crawling engine

Web search engines crawl the web …

The Open Web Search project …

The open-source software …

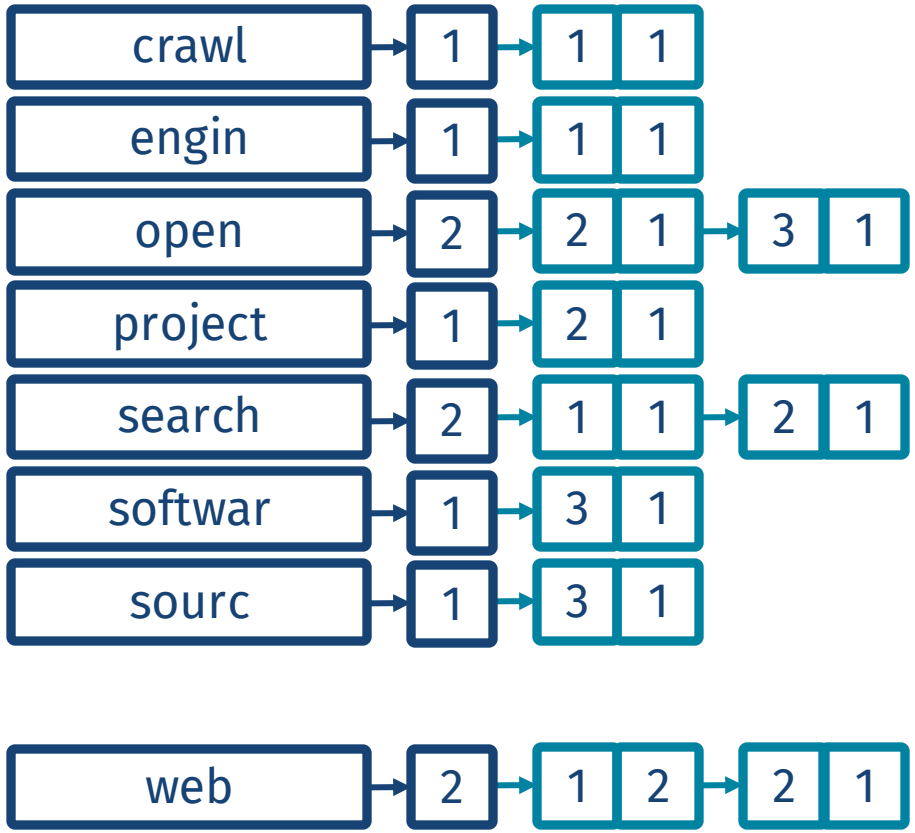
Stemming

crawl engin

Web search engines crawl the web …

The Open Web Search project …

The open-source software …

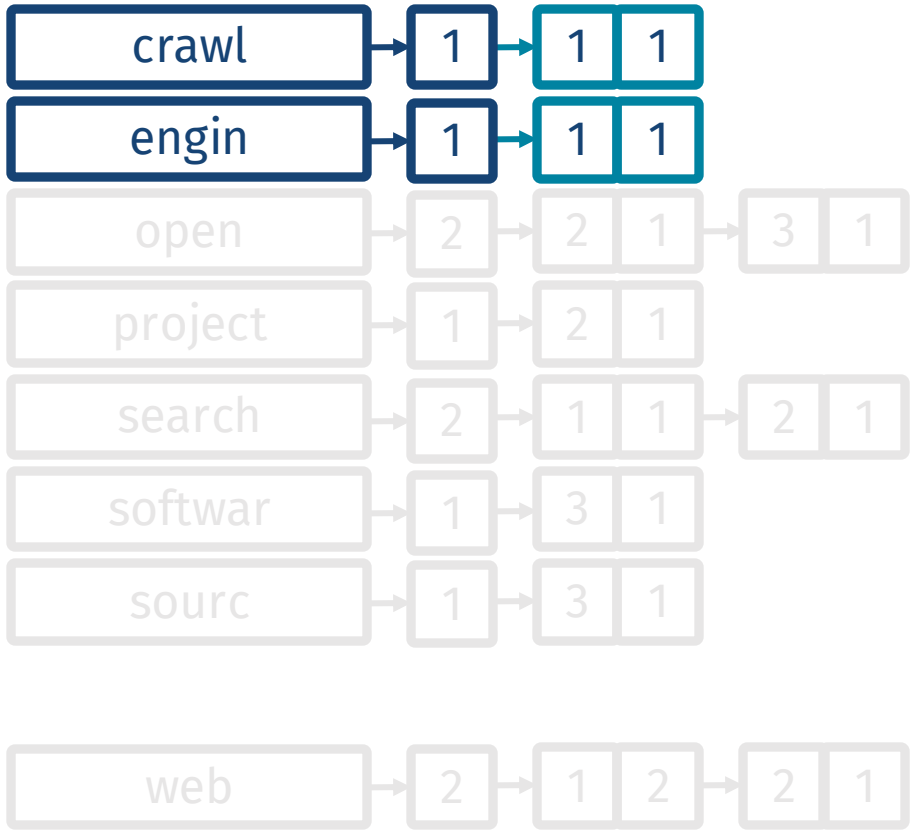


Stemming

Web search engines crawl the web ...

The Open Web Search project ...

The open-source software ...



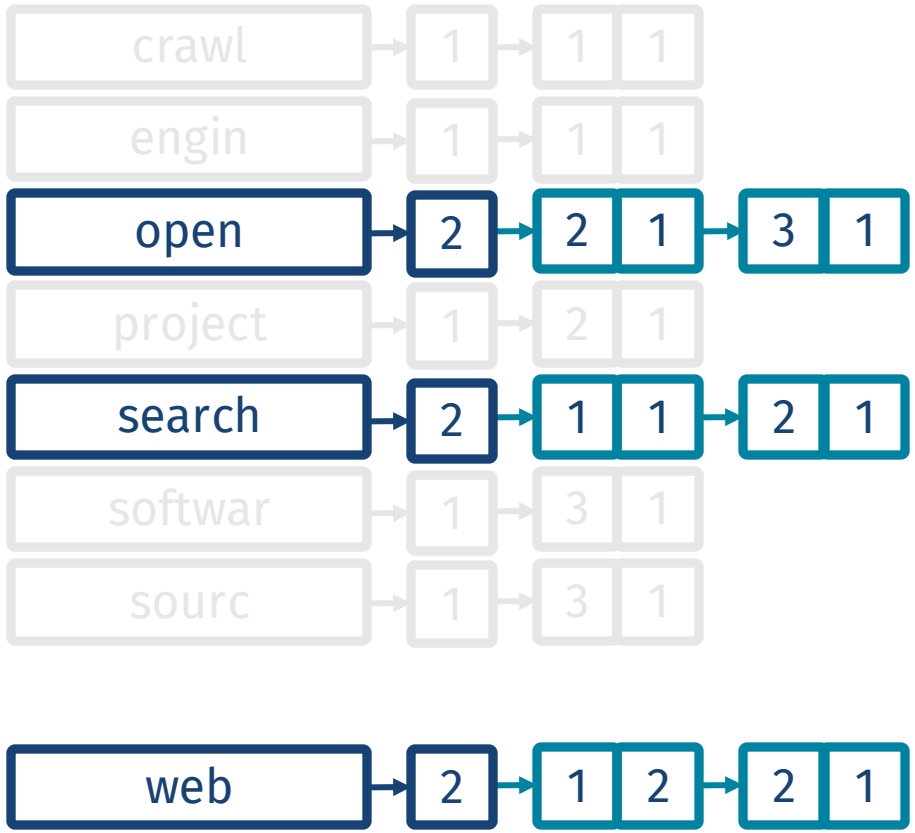
Phrases and proximity

"open web search"

Web search engines crawl the web ...

The Open Web Search project ...

The open-source software ...



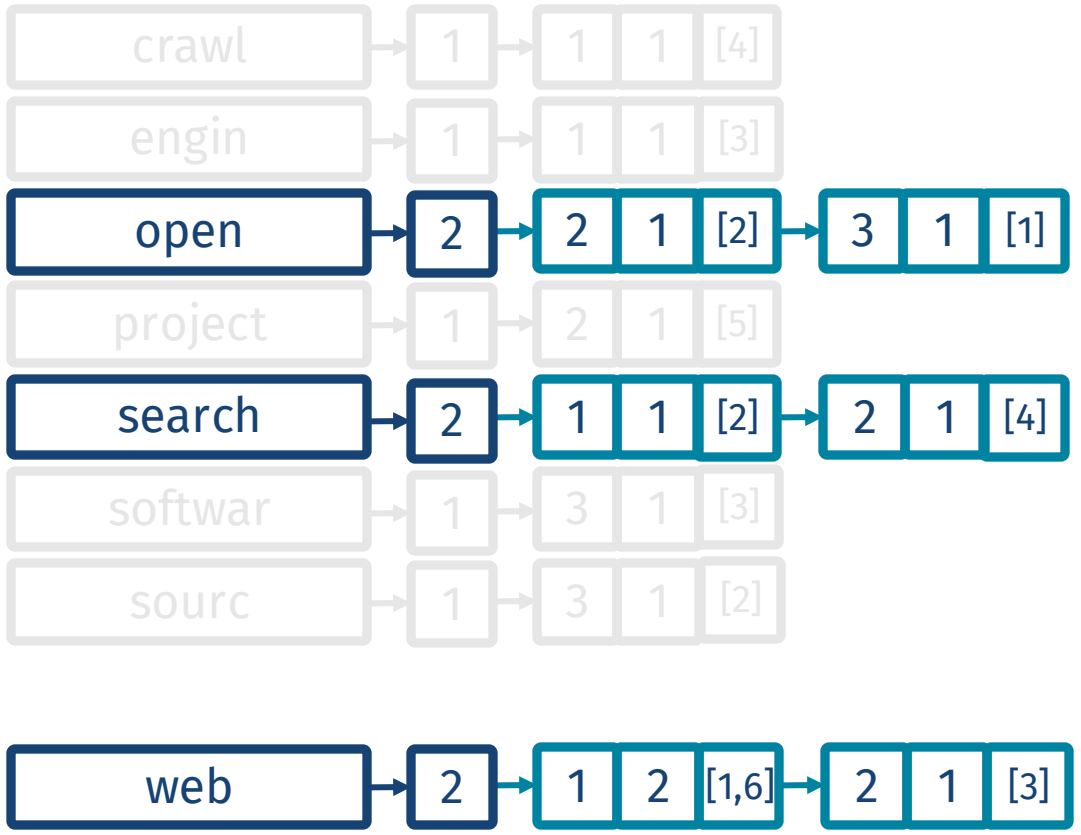
Phrases and proximity

"open web search"

Web search engines crawl the web ...

The Open Web Search project ...

The open-source software ...



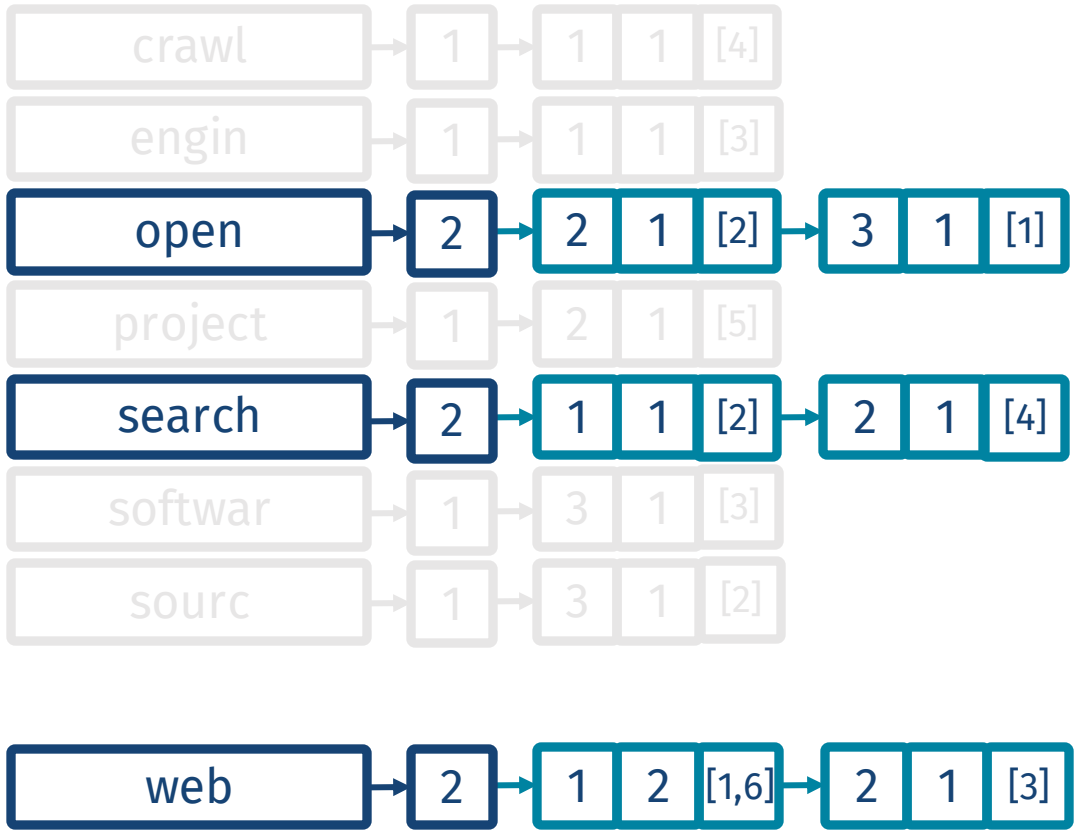
Phrases and proximity

"open web search"

Web search engines crawl the web ...

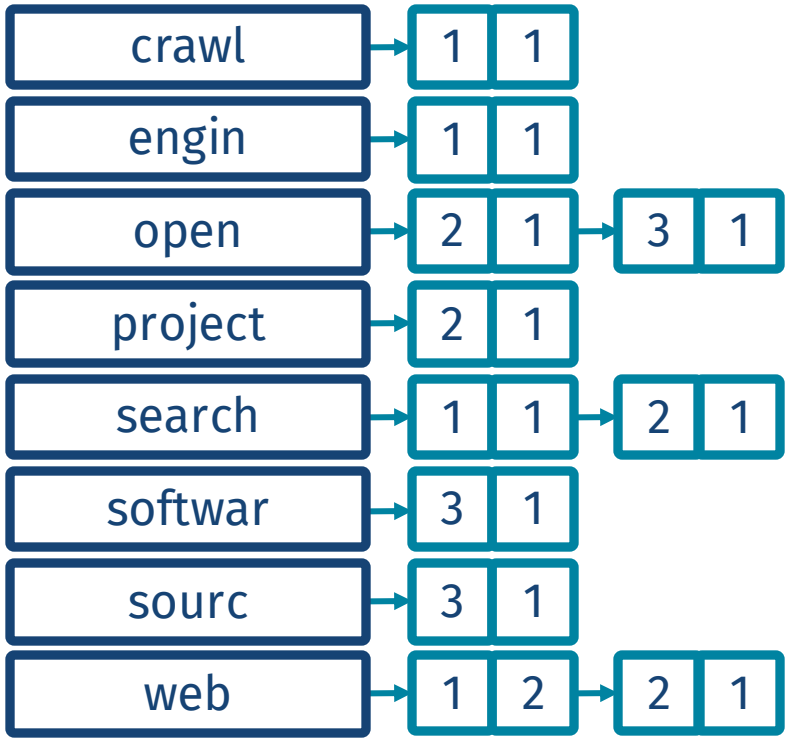
The Open Web Search project ...

The open-source software ...



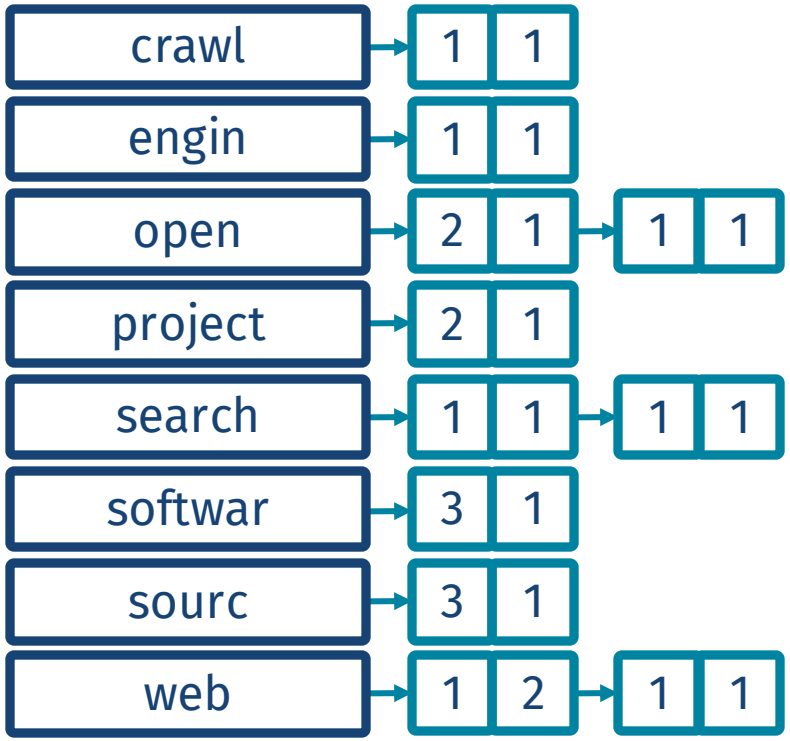
Reducing the size of inverted files

➜ **Delta-gap encoding**
Encode *gaps* between docids, instead of raw docids
- Makes values in posting lists smaller
- Requires posting lists to be sorted



Reducing the size of inverted files

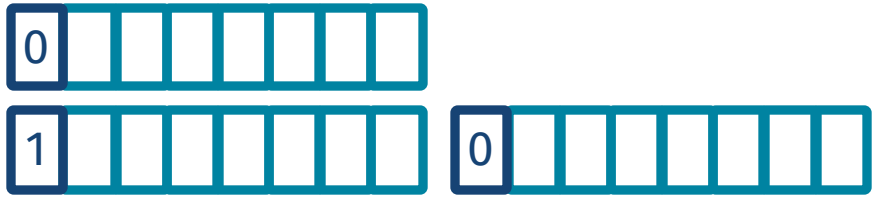
➜ **Delta-gap encoding**
Encode *gaps* between docids, instead of raw docids
- Makes values in posting lists smaller
- Requires posting lists to be sorted

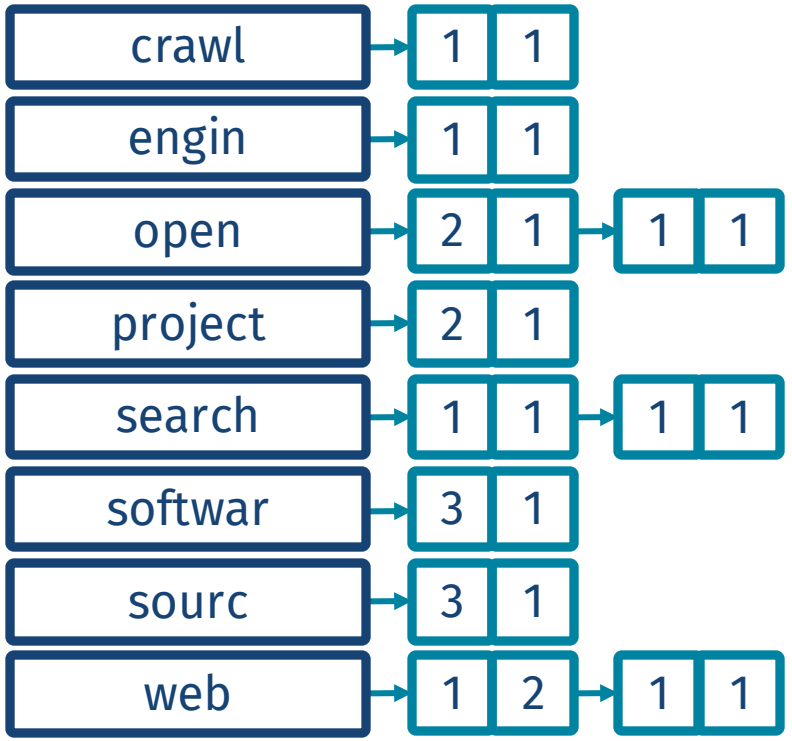


Reducing the size of inverted files

➜ **Delta-gap encoding**
Encode *gaps* between docids, instead of raw docids
- Makes values in posting lists smaller
- Requires posting lists to be sorted

➜ **Compression**
Bit-aligned or byte-aligned
E.g. VByte: use fewer bytes for smaller numbers
- 1 "continuation" bit, 7 data bits





Index shards

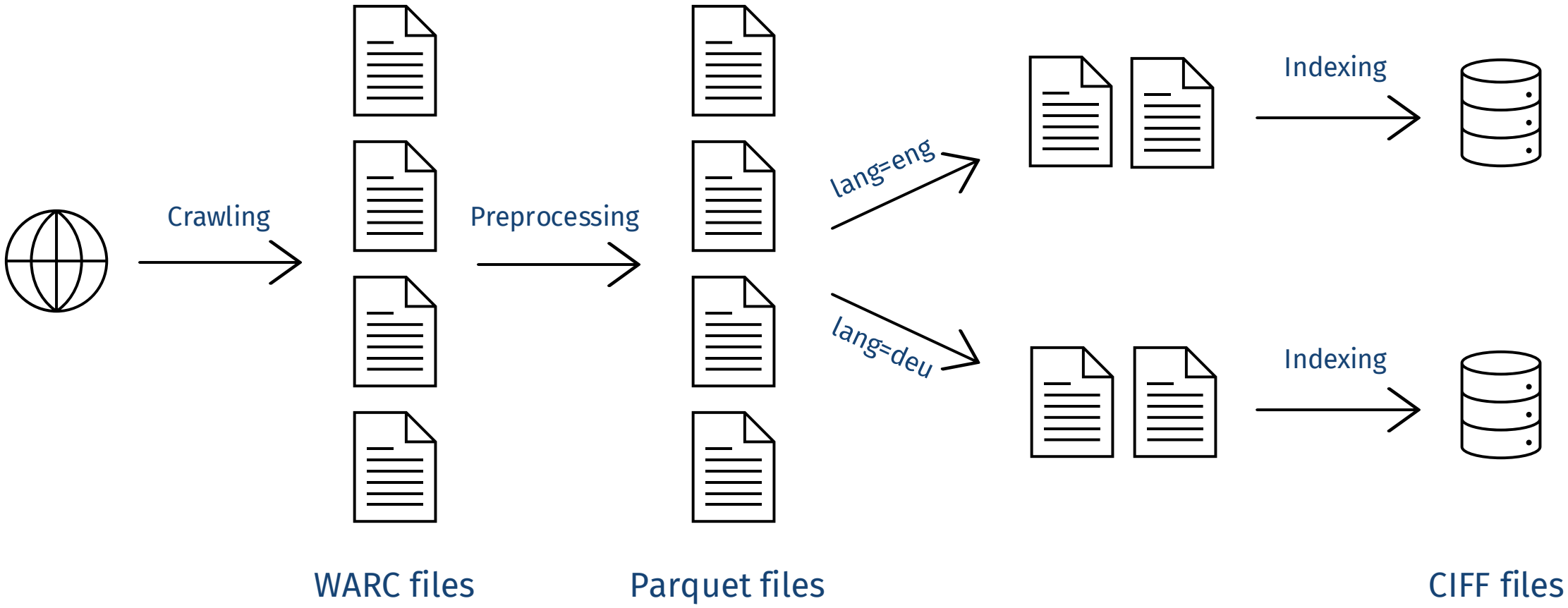
- Inverted files for a subset of the data, e.g.:
  - Topic ("all documents related to sports")
  - Language ("all English documents")
  - Date ("all documents from March 10th")

- Use one or multiple shards for search

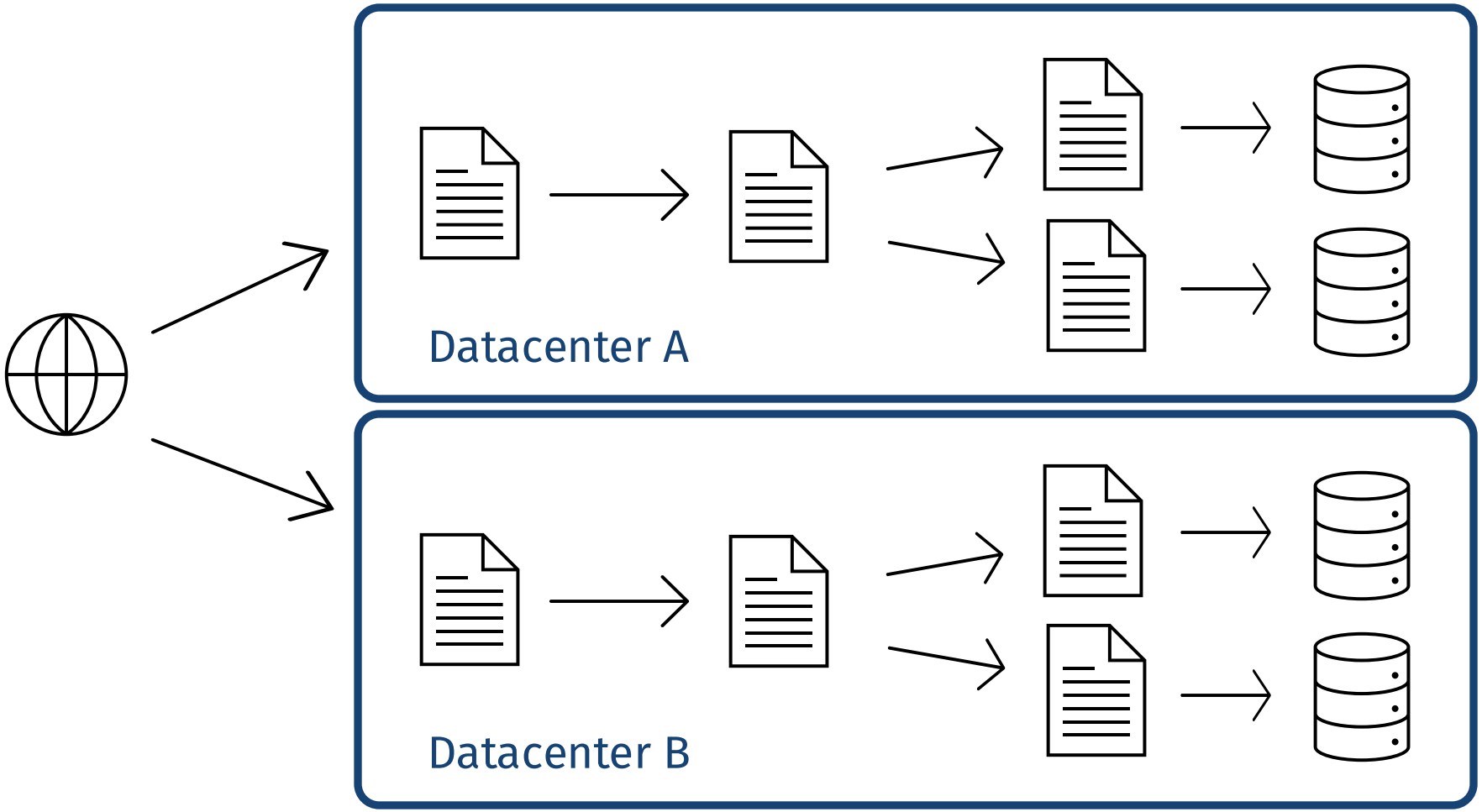
Index shards: research directions

- "Selective Search"
  - Use a few shards for a single search query
- Session search
  - Use a few shards for a full search session
- Personal search
  - Use a few shards for each individual user
- Shard map quality estimation
  - How well clustered are relevant documents for a query?

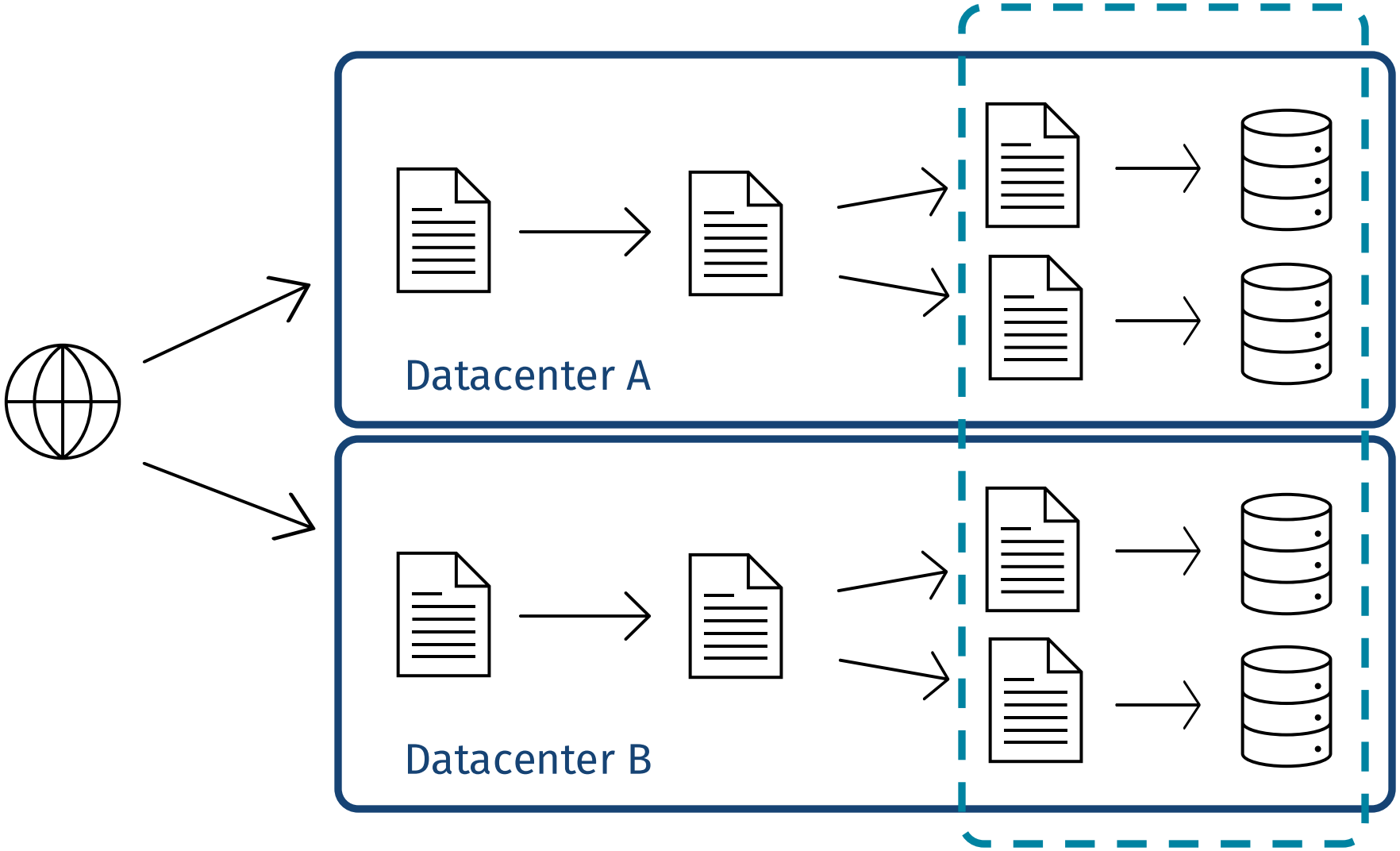
Inverted files in the Open Web Index



Inverted files in the Open Web Index

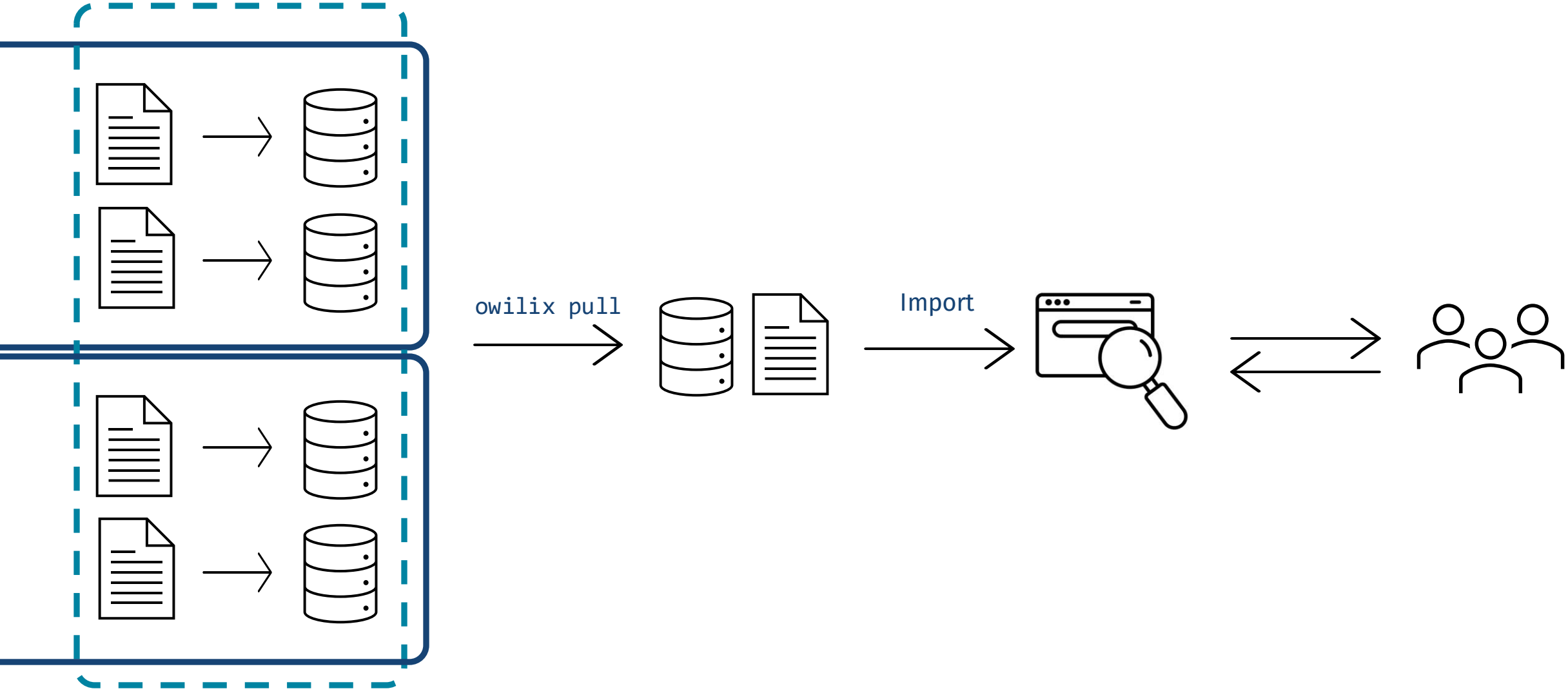


Inverted files in the Open Web Index



The Open Web Index

Usage of the Open Web Index



Distributing index shards:
The Common Index File Format (CIFF)

➜ Binary (protobuf), minimal file format for inverted files

Consists of:

➜ Header, with global statistics (e.g. number of documents, average length)

➜ Posting lists, with term frequencies (no positions, delta-gap encoding + VByte)

➜ Document records, with internal ID, external ID and document length

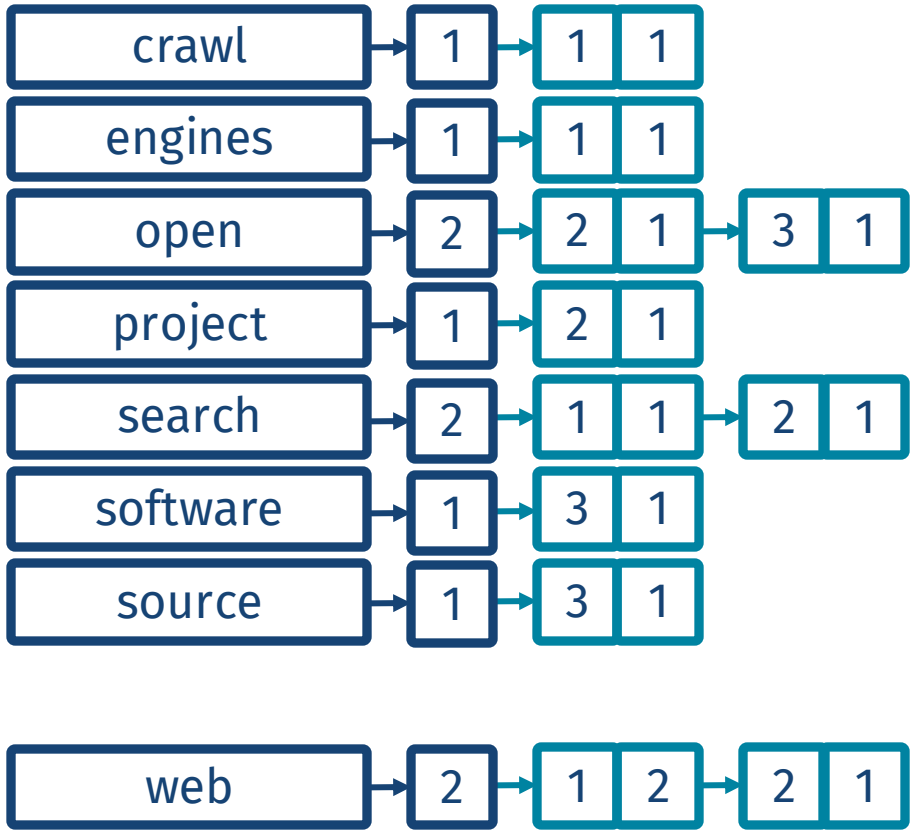
Challenges of index exchange: stopwords

"the Web"

Web search engines crawl the web ...

The Open Web Search project ...

The open-source software ...

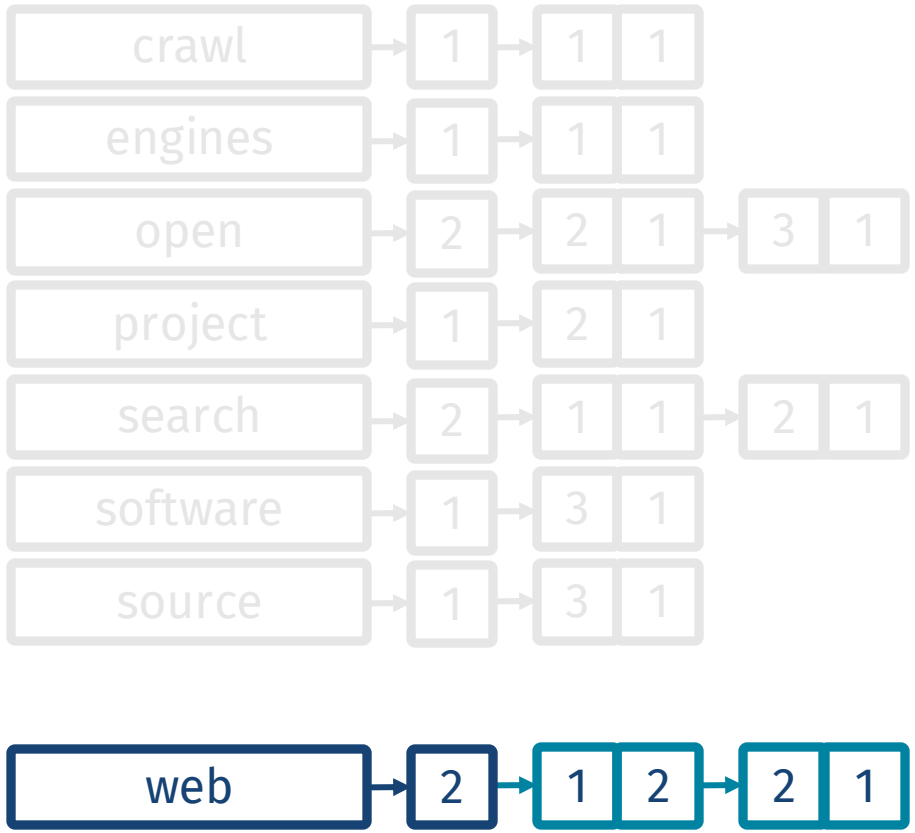


Challenges of index exchange: stopwords

Web search engines crawl the web ...

The Open Web Search project ...

The open-source software ...

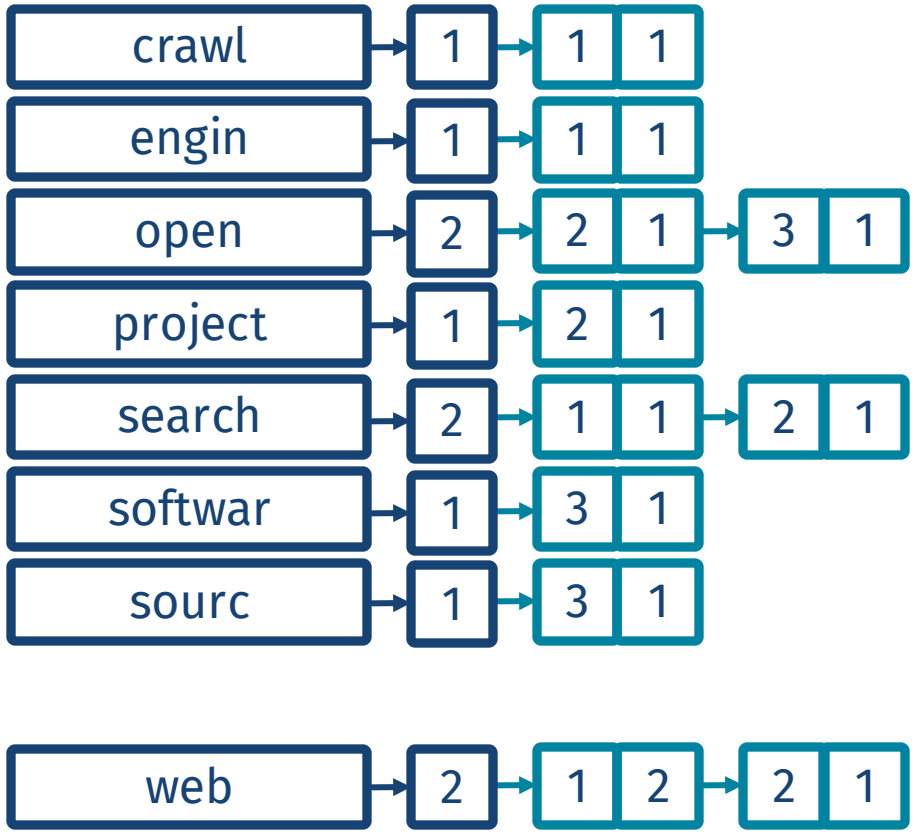


Challenges of index exchange: stemming

Web search engines crawl the web …

The Open Web Search project …

The open-source software …



Challenges of index exchange: stemming

Web search engines crawl the web ...

The Open Web Search project ...

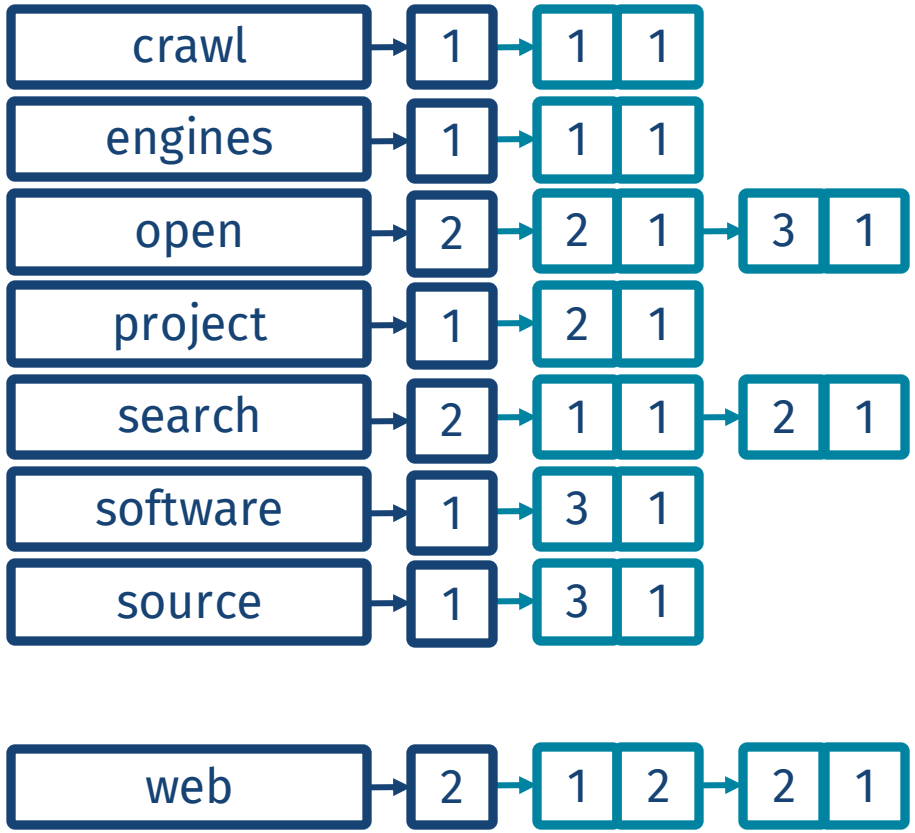
The open-source software ...

Challenges of index exchange: tokenization

Web search engines crawl the web …

The Open Web Search project …

The open-source software …



Challenges of index exchange: tokenization

Web search engines crawl the web ...

The Open Web Search project ...

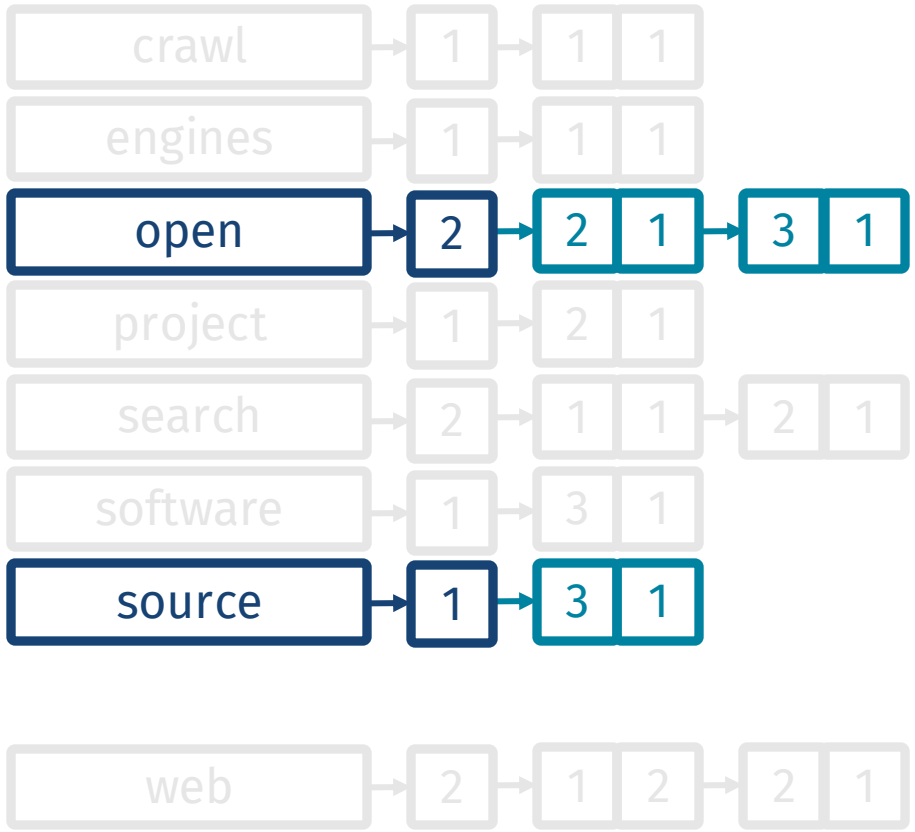
The open-source software ...

Challenges of index exchange: tokenization

Web search engines crawl the web …

The Open Web Search project …

The open-source software …



Column stores as inverted files

Web search engines crawl the web …

The Open Web Search project …

The open-source software …

dict

termid	term	df
0	crawl	1
1	engin	1
2	open	2
3	project	1
4	search	2
5	softwar	1
6	sourc	1
7	web	2

postings

termid	docid	tf
0	0	1
1	0	1
2	1	1
2	2	1
3	1	1
4	0	1
4	1	1
5	2	1
6	2	1
7	0	2
7	1	1

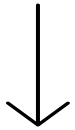
docs

docid	name	len
0	d1	6
1	d2	5
2	d3	3

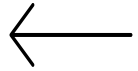
stats

num_docs
3

Column stores as inverted files



docid	score
0	0.45
1	0.41



dict

termid	term	df
0	crawl	1
1	engin	1
2	open	2
3	project	1
4	search	2
5	softwar	1
6	sourc	1
7	web	2

docs

docid	name	len
0	d1	6
1	d2	5
2	d3	3

postings

termid	docid	tf
0	0	1
1	0	1
2	1	1
2	2	1
3	1	1
4	0	1
4	1	1
5	2	1
6	2	1
7	0	2
7	1	1

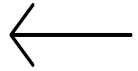
stats

num_docs
3

Column stores as inverted files


```
SELECT
    docid,
    bm25_score(tf, df, len) AS score
FROM postings
    NATURAL JOIN dict
    NATURAL JOIN docs
WHERE term IN ('web', 'search')
GROUP BY docid
ORDER BY score DESC;
```

docid	score
0	0.45
1	0.41



dict

termid	term	df
0	crawl	1
1	engin	1
2	open	2
3	project	1
4	search	2
5	softwar	1
6	sourc	1
7	web	2

postings

termid	docid	tf
0	0	1
1	0	1
2	1	1
2	2	1
3	1	1
4	0	1
4	1	1
5	2	1
6	2	1
7	0	2
7	1	1

docs

docid	name	len
0	d1	6
1	d2	5
2	d3	3

stats

num_docs
3

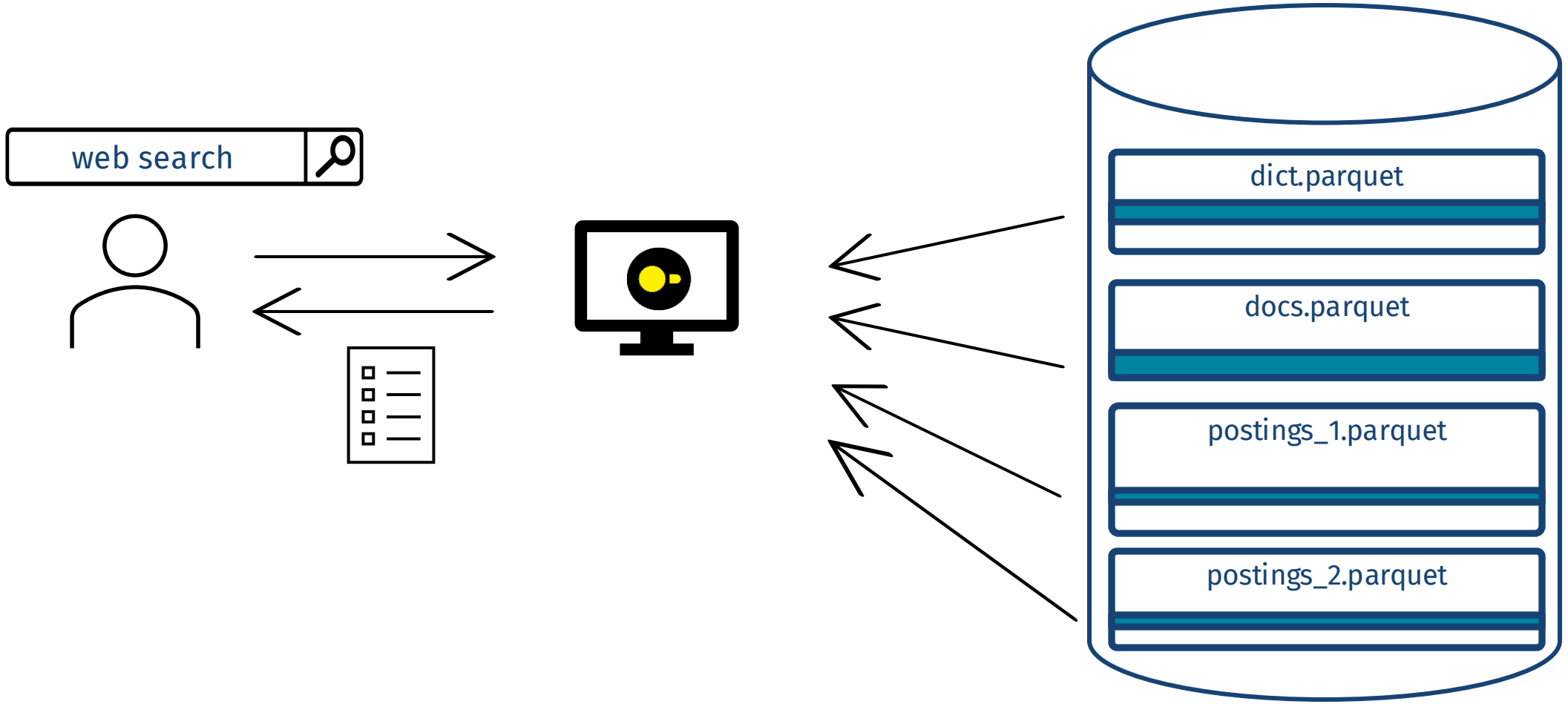
Parquet files

- Column-oriented file format designed for efficient storage and retrieval
- Read only parts (columns, row groups) of the file that you actually need
- Efficient compression
- Supported natively by certain query engines (e.g. DuckDB, Apache Spark)

Simple Storage Service (S3)

- Object storage service (e.g. MinIO, AWS)
- Files are grouped in "buckets"
- Simple HTTP API, with support for RANGE requests to read files partially
- Supported natively by certain query engines (e.g. DuckDB, Apache Spark)

Putting it all together: Direct access to the Open Web Index



Putting it all together: Direct access to the Open Web Index

Advantages

➜ Client-side processing: no need to set up search engine or API

➜ Transparent access to slices/shards of the Open Web Index

Disadvantages

➜ Latency is higher than a search engine with a local index

➜ Object storage (S3) might become a bottleneck (server load, bandwidth, etc.)

Conclusions

Inverted files

- ➜ Efficient structures for term-based search
- ➜ Construction and usage of inverted files
- ➜ Optimizing the size of inverted files

Direct index access

- ➜ Store inverted files in columnar format in object storage
- ➜ Use a client-side engine to directly query the index

The Open Web Index

- ➜ A distributed data set with (daily) index shards
- ➜ To use: "pull" and import index shards

Questions?