Information Retrieval

Data Storage

**Ilya Markov**

i.markov@uva.nl

University of Amsterdam

Course overview

Offline

| Data Acquisition | $\Rightarrow$ | Data Processing | $\Rightarrow$ | Data Storage |

Online

| Query Processing | $\Rightarrow$ | Ranking | | Evaluation |

Advanced

| Advanced topics in IR and Web Search |

This lecture

Offline: Data Acquisition → Data Processing → **Data Storage**

Online: Query Processing → Ranking, Evaluation

Advanced: Advanced topics in IR and Web Search
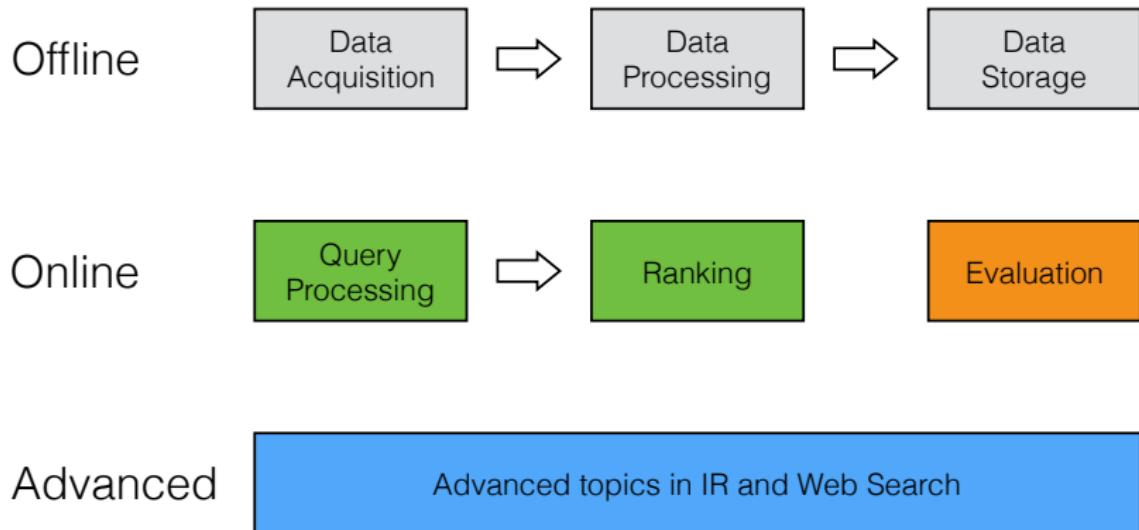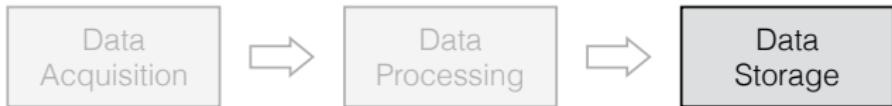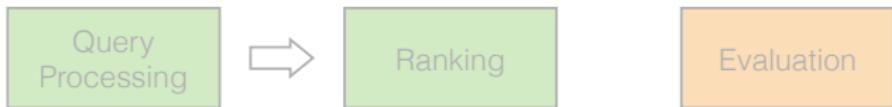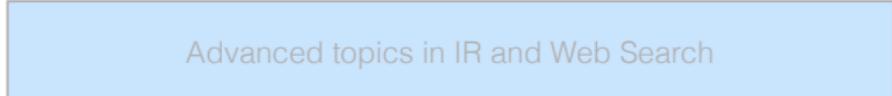
Data storage methods

- File
- File system
- Database
- **Index**

Outline

1. Basic indexing architecture

2. Inverted index

3. Constructing an index

4. Updating an index

5. Compressing an index

6. Partitioning an index

7. Summary

Outline

No merge

Memory

Delta index

. . .

Delta index

Old main index

Disk
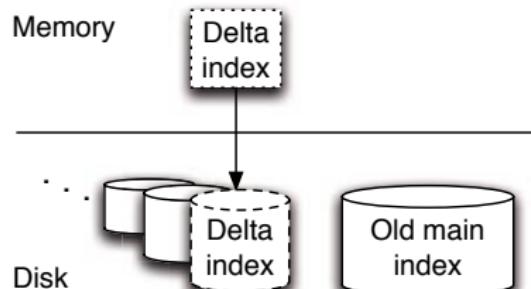
- Low index maintenance cost
- High query processing cost

B. Cambazoglu and R. Baeza-Yates, "Scalability Challenges in Web Search Engines"

Incremental update
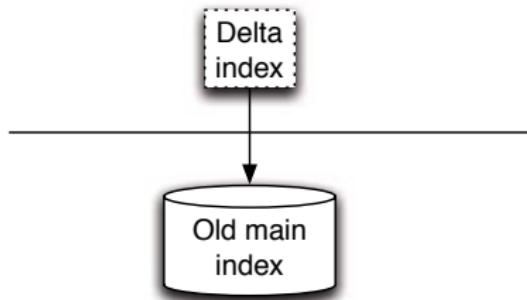
- Keeps free buffer space
- No read/write of entire index when updating
- Inverted lists are accessed concurrently
- Run out of free buffer space

B. Gambazoglu and R. Baeza-Yates, "Scalability Challenges in Web Search Engines"

Immediate merge

Delta index

Old main index

New main index
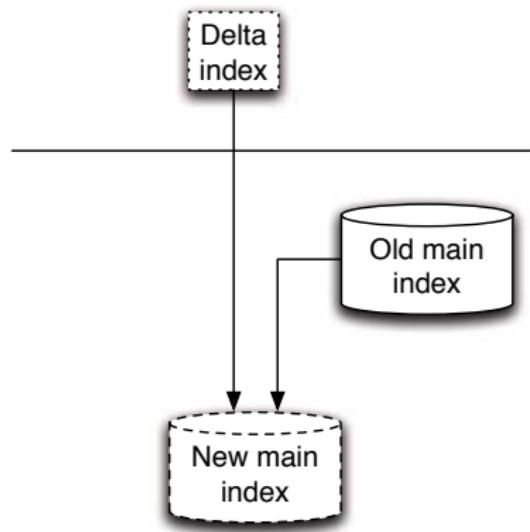
- Always a single index
- Read/write of entire index when updating

B. Cambazoglu and R. Baeza-Yates, "Scalability Challenges in Web Search Engines"

Lazy merge
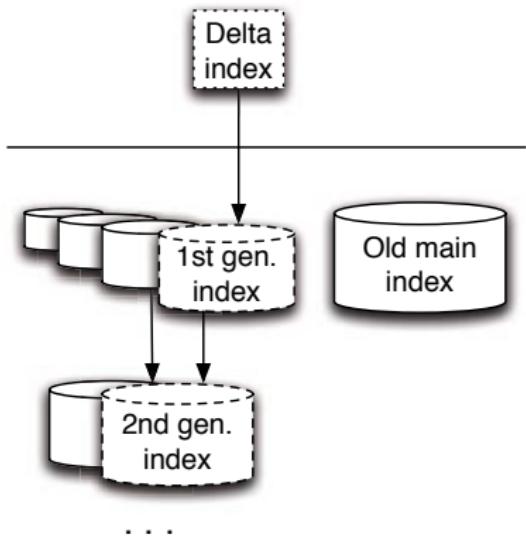
- Trade-off between index maintenance cost and query processing cost

B. Cambazoglu and R. Baeza-Yates, "Scalability Challenges in Web Search Engines"

Page deletions

- Maintain identifiers of deleted documents in memory,
  access during query processing
- Garbage collection

Summary

- Updating strategies
    - No merge
    - Incremental update
    - Immediate merge
    - Lazy merge
- Page deletions

Outline

Gap values

Original values $\left(\begin{array}{llllllll}17 & 18 & 28 & 40 & 44 & 47 & 56 & 58\end{array}\right)$

Gap values $\left(\begin{array}{llllllll}17 & 1 & 10 & 12 & 4 & 3 & 9 & 2\end{array}\right)$
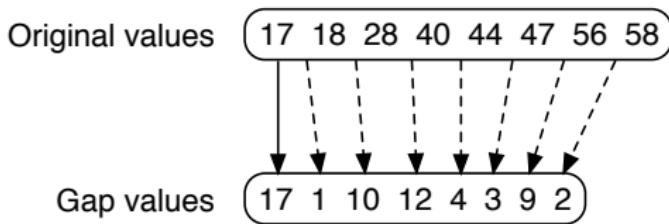
B. Cambazoglu and R. Baeza-Yates, "Scalability Challenges in Web Search Engines"

Compression algorithms

Encoding	Input sequence	Output	Parameters	Encoded values
Unary	gaps	bit-aligned	nonparametric	individual values
Gamma	gaps	bit-aligned	nonparametric	individual values
Delta	gaps	bit-aligned	nonparametric	individual values
Variable byte	gaps	byte-aligned	nonparametric	individual values
Golomb	gaps	bit-aligned	parametric	individual values
Simple-9	gaps	word-aligned	parametric	blocks of values
PForDelta	gaps	bit-aligned	parametric	blocks of values
Binary interpol.	monotonic sequences	bit-aligned	parametric	bisections
Elias-Fano	monotonic sequences	bit-aligned	parametric	entire sequence

B. Cambazoglu and R. Baeza-Yates, "Scalability Challenges in Web Search Engines"

Reassigning document identifiers

Document
identifier
mapping

$1 \longrightarrow 1$

$2 \longrightarrow 9$

$3 \longrightarrow 2$

$4 \longrightarrow 7$

$5 \longrightarrow 8$

$6 \longrightarrow 3$

$7 \longrightarrow 5$

$8 \longrightarrow 6$

$9 \longrightarrow 4$

Original inverted lists

L1: 1 3 6 8 9   L2: 2 4 5 6 9   L3: 3 6 7 9

Original gap values

L1: 2 3 2 1   L2: 2 1 1 3   L3: 3 1 2

Reordered inverted lists

L1: 1 2 3 4 6   L2: 3 4 7 8 9   L3: 2 3 4 5

New gap values

L1: 1 1 1 2   L2: 1 3 1 1   L3: 1 1 1

B. Cambazoglu and R. Baeza-Yates, "Scalability Challenges in Web Search Engines"

Outline

Document-based
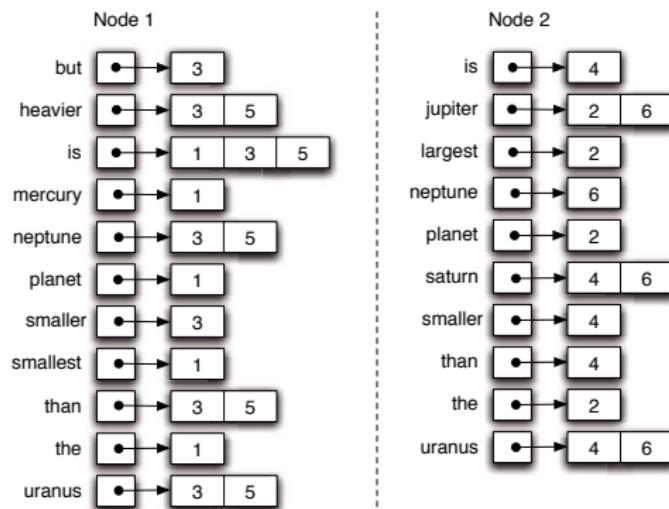
- More balanced
- Multiple disk accesses for reading a list
- Faster disk access
- Vocabulary needs to be replicated
- Simple index construction

B. Cambazoglu and R. Baeza-Yates, "Scalability Challenges in Web Search Engines"

Term-based
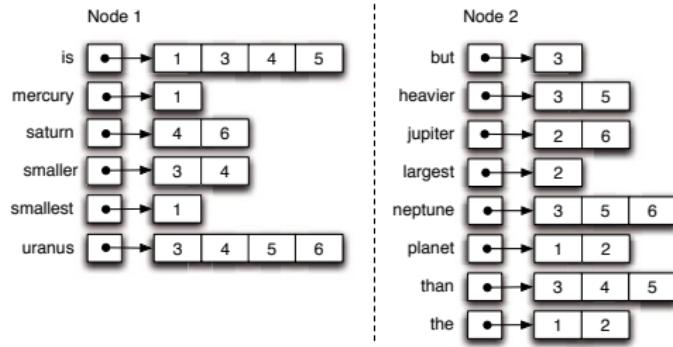
- Less balanced
- Single disk access for reading a list
- Page attribute file needs to be replicated
- Coordination and communication during index construction

B. Cambazoglu and R. Baeza-Yates, "Scalability Challenges in Web Search Engines"

Comparison

	Index partitioning strategies	
Aspect	Document-based	Term-based
Vocabulary	Replicated on each node – $O(KN)$	Partitioned over the nodes – $O(N)$
Page attribute file	Partitioned over the nodes – $O(M)$	Replicated on each node – $O(KM)$
List access	Many concurrent partial list accesses – $O(K)$	Single list access – $O(1)$
List I/O from the disk	Shorter lists – takes less time to read	Longer lists – takes more time to read
Index construction	Simpler – local indexes	More complex – global index
Index maintenance	Updates on a single node – $O(1)$	Updates on many nodes – $O(K)$

- K – number of nodes
- N – number of terms
- M – number of pages

B. Cambazoglu and R. Baeza-Yates, "Scalability Challenges in Web Search Engines"

Outline

Outline

1. Basic indexing architecture

2. Inverted index

3. Constructing an index

4. Updating an index

5. Compressing an index

6. Partitioning an index

7. Summary

Materials

- Croft et al., Chapter 5
- Manning et al., Chapters 1.2–1.3, 2.3–2.4
- B. Barla Cambazoglu and Ricardo Baeza-Yates
  Scalability Challenges in Web Search Engines
  Morgan & Claypool Publishers, 2017

Course overview

Offline

| Data Acquisition | $\Rightarrow$ | Data Processing | $\Rightarrow$ | Data Storage |

Online

| Query Processing | $\Rightarrow$ | Ranking | | Evaluation |

Advanced

| Advanced topics in IR and Web Search |

Next lecture

Offline	Data Acquisition	⇒	Data Processing	⇒	Data Storage
Online	Query Processing	⇒	Ranking		Evaluation
Advanced	Advanced topics in IR and Web Search				