BACS3183

Advanced Database Management

Chapter 6
Indexing

Learning Outcomes

At the end of this class, you should be able to

- Create indexes with SQL.

- Compare performance in retrieving information when indices are used as to when they are not used.

- Explain the role of an inverted index in locating a document in a collection.

1. Indexes

- **Indexes are special lookup data structures that the search engine can use to speed up data retrieval.**

- **Similar to**
  - ✓ **book index at the back of a book**
  - ✓ **book catalog in library**

Linear search

Large number of blocks holding records
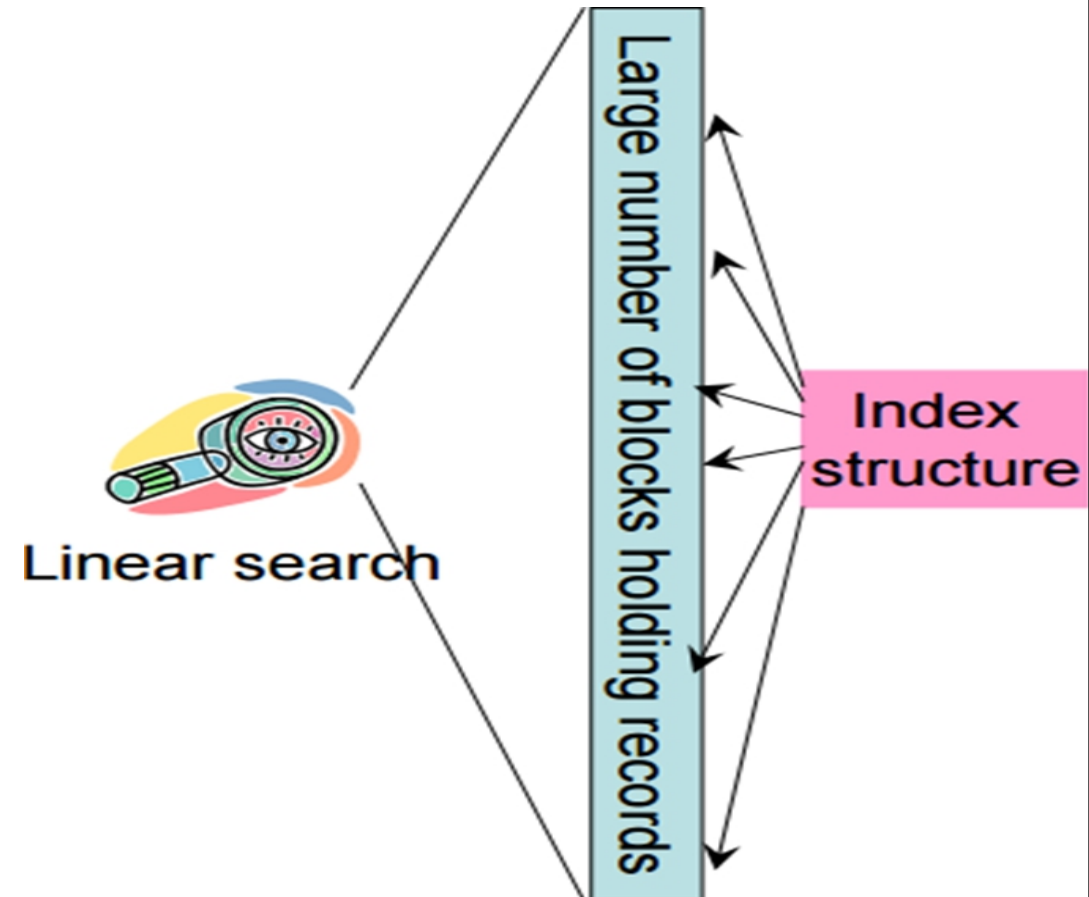
Index structure

Index Definition in SQL

- **Create an index in Oracle**

  CREATE [UNIQUE] INDEX *indexName*

    ON *TableName* (*columnName* [ASC | DESC])

 E.g.:
 CREATE INDEX *state_ndx_index* ON *Customer (cus_state)*

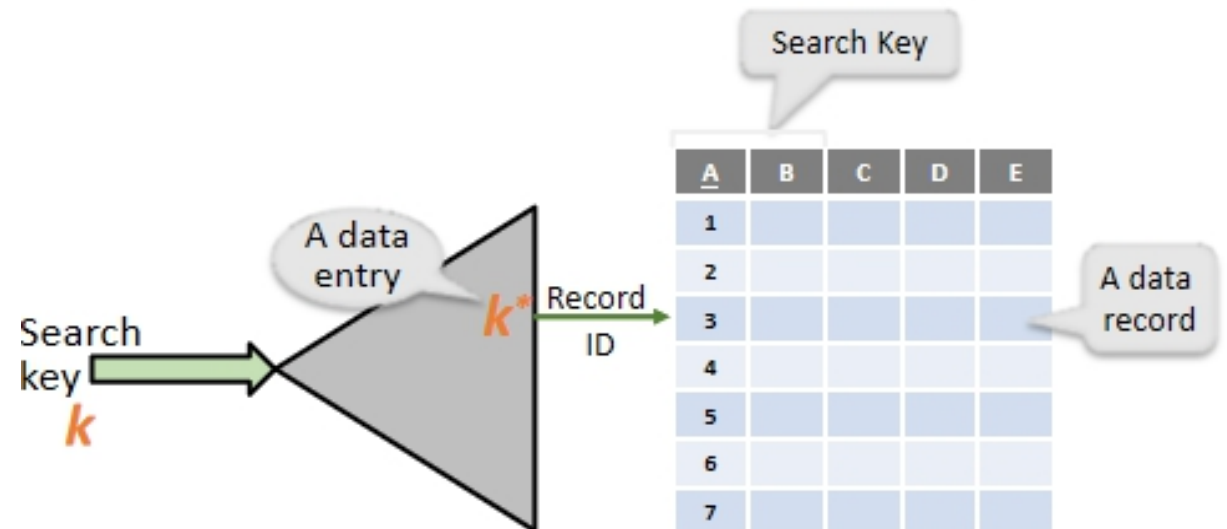- **To drop an index**

  DROP INDEX *<index-name>*
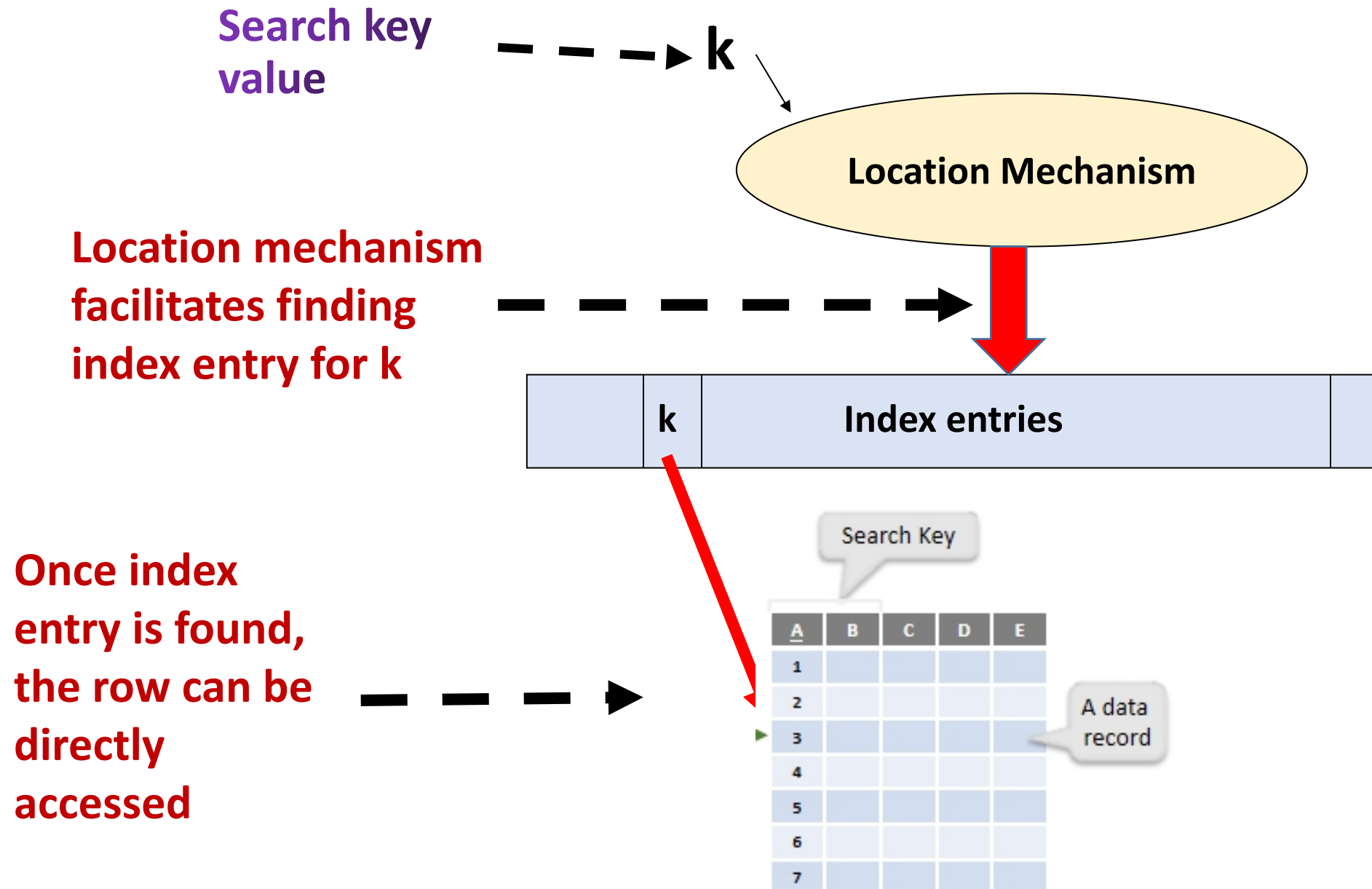
Index Structure

An index contains a collection of *entries*, and supports **efficient retrieval** of all entries with a given search key value k.

To locate (one or more) data records with search key value k

- Search the index using k to find the desired index entry $k*$
- The **index entry $k*$ contains information to locate** (one or more) **data records** with search key value k

Index Structure

Search key value $\dashrightarrow$ k

Location Mechanism

Location mechanism facilitates finding index entry for k $\dashrightarrow$

| | k | Index entries | |

Once index entry is found, the row can be directly accessed $\dashrightarrow$

Search Key

A	B	C	D	E
1				
2				
3				
4				
5				
6				
7				

A data record

2. Indexes and Query Performance

Database System: Design, Implementation & Application by Rob & Colonel

SELECT CUS_NAME, CUS_STATE FROM CUSTOMER WHERE CUS_STATE = 'FL';

Requires a total of 10 I/O accesses (5 accesses to the index and 5 accesses to the data) ie the DBMS would save approximately 14,776 I/O requests

Indexes and Query Performance

- If indexes are so important, why not index every column in every table?

- **Not practical**
  - taxes the DBMS too much in terms of **index-maintenance processing**, especially if the table
    - ✓ has many **attributes**
    - ✓ has many **rows**
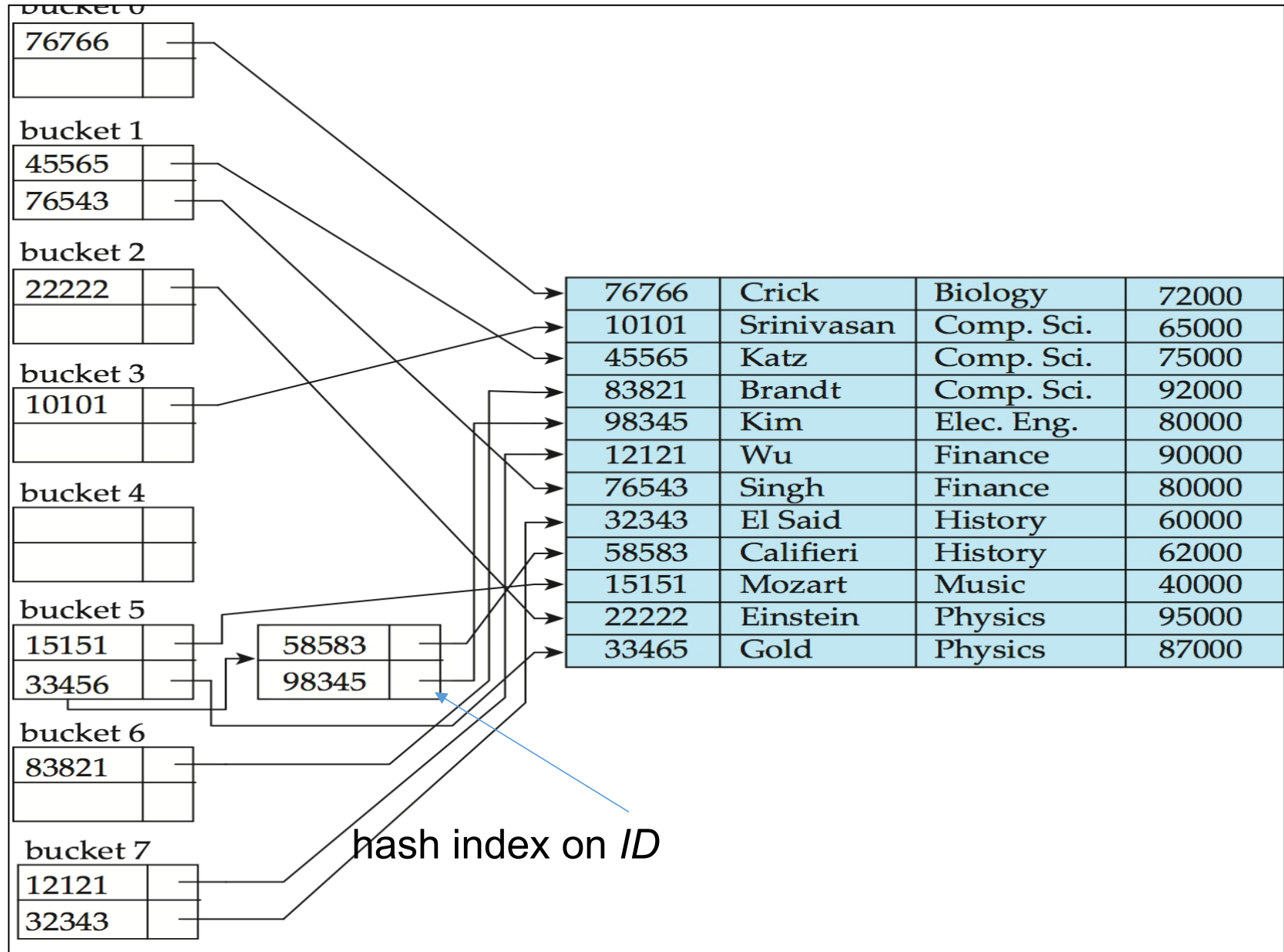   - ✓ and/or requires **many inserts, updates, and/or deletes.**

Indexing Structures

- **Common structures used**
  - *Hash indexes*
  - *B-tree indexes*
  - *Bitmap indexes*
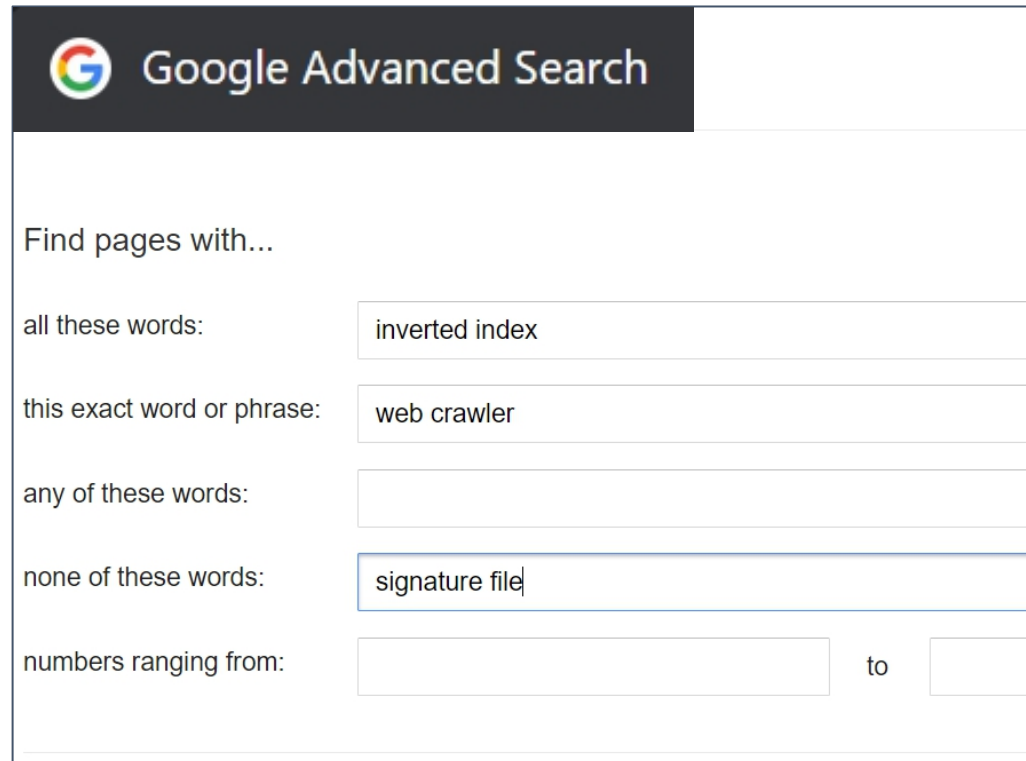

CUSTOMER TABLE

CUS_ID	CUS_LNAME	CUS_FNAME	CUS_PHONE	REGION_CODE
12	Adams	Charlie	4555	NW
23	Blair	Robert	5426	SE
37	Coronel	Carlos	2358	SW
43	DePrince	Albert	6543	NE
55	Greer	Tim	2764	SE
58	Kyle	Ruben	2453	SW
62	Lee	John	7895	NE
65	Maier	Jerry	7689	NW
68	Morris	Steve	4588	NW
72	Rob	Pete	8123	NE
75	Sarver	Lee	6193	SE
80	Strickland	Tomas	3120	SW
82	Timmons	Douglas	3499	NE

B-tree Index
On CUS_LNAME

Bitmap Index
On REGION_CODE

In the bitmap index, each bit represents one region code. In the first row, bit number two is turned on, thus indicating that the first row region code value is NW

REGION_CODE = 'NW'

Region →	NE	NW	SE	SW		
	Bit 1	Bit 2	Bit 3	Bit 4	Bit ...	Bit
	0	1	0	0		
	0	0	1	0		
	0	0	0	1		
	1	0	0	0		
	0	0	1	0		
	0	0	0	1		
	1	0	0	0		
	0	1	0	0		
	0	1	0	0		
	1	0	0	0		
	0	0	1	0		
	0	0	0	1		
	1	0	0	0		

Database System: Design, Implementation & Application by Rob & Colonel

Hash Index

hash index on *ID*

3. Full-text Database Systems

- **For storing and accessing document collections such as newspaper archives, research papers, articles etc.**

- **Indexing on text content is needed**

- **Index structure for text database systems must support efficient evaluation of boolean queries.**

Indexing the Web

- **A web search can cover at least one billion documents**

- **Web search engines can serve hundreds or thousands of queries per second**

- **Use an index is to increase the speed and efficiency of searches of the document collection**

- **Most search engines use inverted index**

▶ Figure 19.7   The various components of a web search engine.

https://www.deepcrawl.com/knowledge/technical-seo-library/search-engine-indexing/
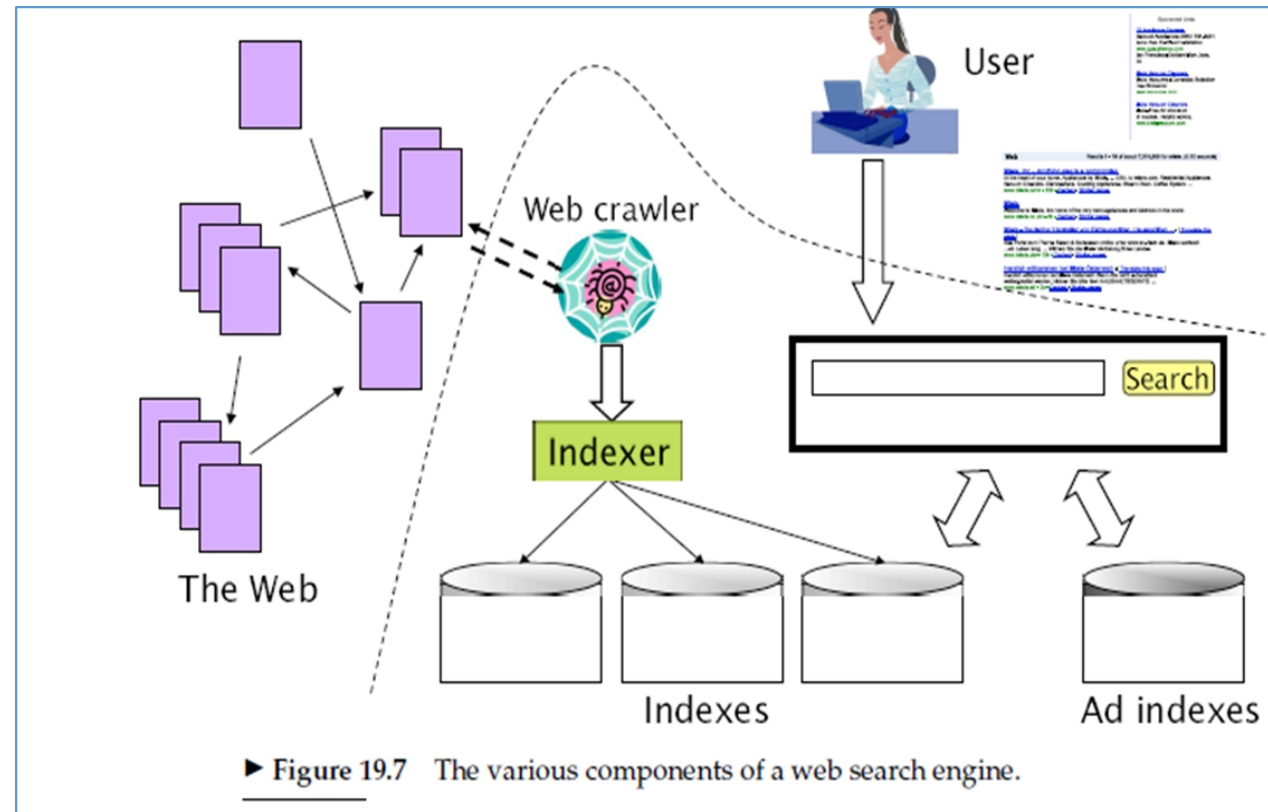
Major Text Indexing Methods

- **Inverted index**
  - **Effective for very large collections of documents**
  - **Use in most search engines**

- **Signature files**
  - **Usually not used for large database sizes**

- **Suffix trees and arrays**

Inverted index

- **The inverted index structure is composed of :**
  - **vocabulary**
  - **occurrences**

- **The vocabulary is the set of all different words in the text**

- **For each word, the index stores the document ID and the occurrences in which that word appears.**

- **A query with a conjunction of several terms is evaluated by retrieving the inverted lists of the query terms one at a time and intersecting them.**
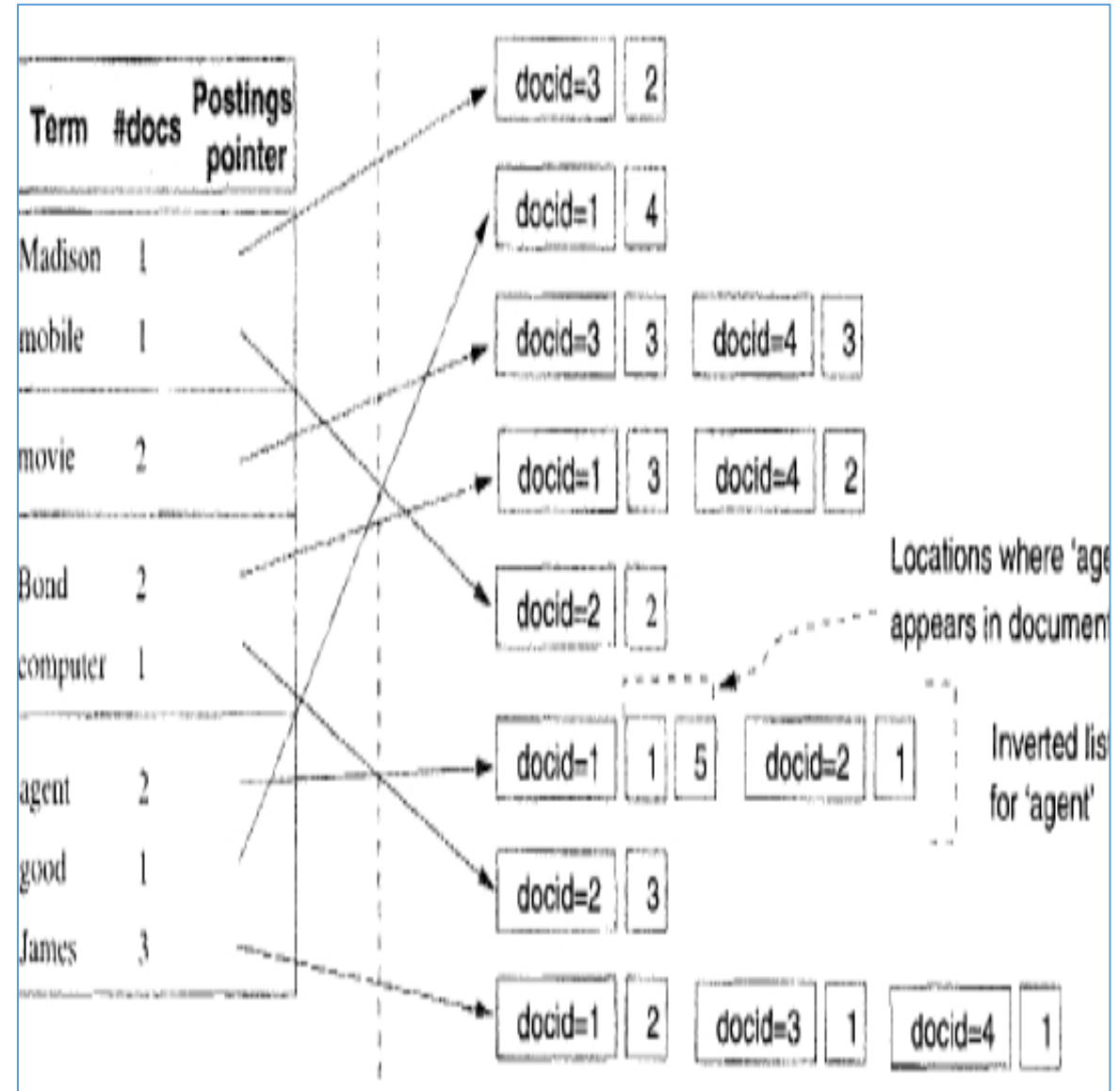
References

Database Systems: A Practical Approach to Design, Implementation and Management.
Connolly, T. M. and Begg, C. E.

Modern Database Management.

Hoffer, J.A., Prescott, M., and McFadden,F.

Database System Concepts

 Silberschatz, A., Korth, H,. and Sudarshan, S.

Database Processing Fundamentals, Design & Implementation.

Kroenke D.