

File Organizations and Indexing

(From Chapter 8)

Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Files: Desired Operations

- Insert record
- Delete record
- Scan records
- Equality search
- Range search

Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Alternative File Organizations

Many alternatives exist, *each ideal for some situation, and not so good in others:*

- [Heap files](#)
- [Sorted Files](#)
- [Hashed Files](#)

Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Cost Model for Our Analysis

We ignore CPU costs, for simplicity:

- **B:**
- **R:**
- **D:**

☒ *Good enough to show the overall trends!*

Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Assumptions in Our Analysis

- Single record insert and delete.
- Heap Files:
 - Equality selection on key; exactly one match.
 - Insert always at end of file.
- Sorted Files:
 - Files compacted after deletions.
 - Selections on sort field(s).
- Hashed Files:
 - No overflow buckets, 80% page occupancy.

Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Cost of Operations

	Heap File	Sorted File	Hashed File
Scan all recs	BD	BD	1.25 BD
Equality Search	0.5 BD	D log₂B	D
Range Search	BD	D (log₂B + # of pages with matches)	1.25 BD
Insert	2D	Search + BD	2D
Delete	Search + D	Search + BD	2D

☒ *Several assumptions underlie these (rough) estimates!*

Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Indexes

- An *index* on a file speeds up selections on the *search key fields* for the index.

Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Alternatives for Data Entry k^* in Index

- Three alternatives:

- Choice of alternative for data entries is orthogonal to the indexing technique used

Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Alternatives for Data Entries (Contd.)

- Alternative 1:

Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Alternatives for Data Entries (Contd.)

- Alternatives 2 and 3:

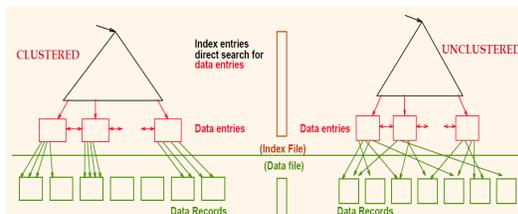
Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Index Classification

- *Primary vs. secondary:*
 - *Unique index:*
- *Clustered vs. unclustered:*

Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Clustered vs. Unclustered Index



Database Management Systems, R. Ramakrishnan and Johannes Gehrke

Summary

- Many alternative file organizations exist, each appropriate in some situation.
- If selection queries are frequent, sorting the file or building an *index* is important.
 - Hash-based indexes only good for equality search.
 - Sorted files and tree-based indexes best for range search; also good for equality search. (Files rarely kept sorted in practice; B+ tree index is better.)
- Index is a collection of data entries plus a way to quickly find entries with given key values.
