Indexing Structures for Files

Handout 8

Chapter Outline

- Types of Single-level Ordered Indexes

  - Primary Indexes
  - Clustering Indexes
  - Secondary Indexes

- Multilevel Indexes

Indexes as Access Paths

- A single-level index is an auxiliary file that makes it more efficient to search for a record in the data file.
- The index is usually specified on one field of the file (although it could be specified on several fields)
- One form of an index is a file of entries <**field value, pointer to record**>, which is ordered by field value
- The index is called an access path on the field.

Indexes as Access Paths (contd.)

- The index file usually occupies considerably less disk blocks than the data file because its entries are much smaller

- Indexes can be characterized as dense or sparse
  - A **dense index** has an index entry for every search key value (and hence every record) in the data file.

  - A **sparse (or nondense) index**, on the other hand, has index entries for only some of the search values

Types of Single-Level Indexes

- Primary Index

  - The data file is ordered on a **primary key field**

  - Includes one index entry *for each block* in the data file; the index entry has the key field value for the *first record* in the block, which is called the *block anchor*

  - A similar scheme can use the *last record* in a block.

Types of Single-Level Indexes

- Primary Index

  - A primary index is a nondense (sparse) index, since it includes an entry for each disk block of the data file and the keys of its anchor record rather than for every search value.

Types of Single-Level Indexes

- Clustering Index

  - Defined on an ordered data file

  - The data file is ordered on a *non-key field* unlike primary index, which requires that the ordering field of the data file have a distinct value for each record.

Types of Single-Level Indexes

- Clustering Index
  - Includes one index entry *for each distinct value* of the field.
  - The index entry points to the first data block that contains records with that field value.
  - It is another example of *nondense* index where Insertion and Deletion is relatively straightforward with a clustering index.

Types of Single-Level Indexes

- Secondary Index

  - A secondary index provides a secondary means of accessing a file for which some primary access already exists.

  - The secondary index may be on a field which is a **candidate key** and has a unique value in every record, or a **non-key** with duplicate values.

Types of Single-Level Indexes

- The index is an ordered file with two fields.
  - The first field is of the same data type as some **non-ordering field** of the data file that is an indexing field.
  - The second field is either a **block** pointer or a record pointer.
  - There can be *many* secondary indexes (and hence, indexing fields) for the same file.
- Includes one entry *for each record* in the data file; hence, it is a *dense index*

Multi-Level Indexes

- Because a single-level index is an ordered file, we can create a primary index *to the index itself*;
  - In this case, the original index file is called the *first-level index* and the index to the index is called the *second-level index*.

Multi-Level Indexes

- We can repeat the process, creating a third, fourth, ..., top level until all entries of the *top level* fit in one disk block

- A multi-level index can be created for any type of first-level index (primary, secondary, clustering) as long as the first-level index consists of *more than one* disk block

Multi-Level Indexes

- Such a multi-level index is a form of *search tree*

  - However, insertion and deletion of new index entries is a severe problem because every level of the index is an *ordered file*.

Summary

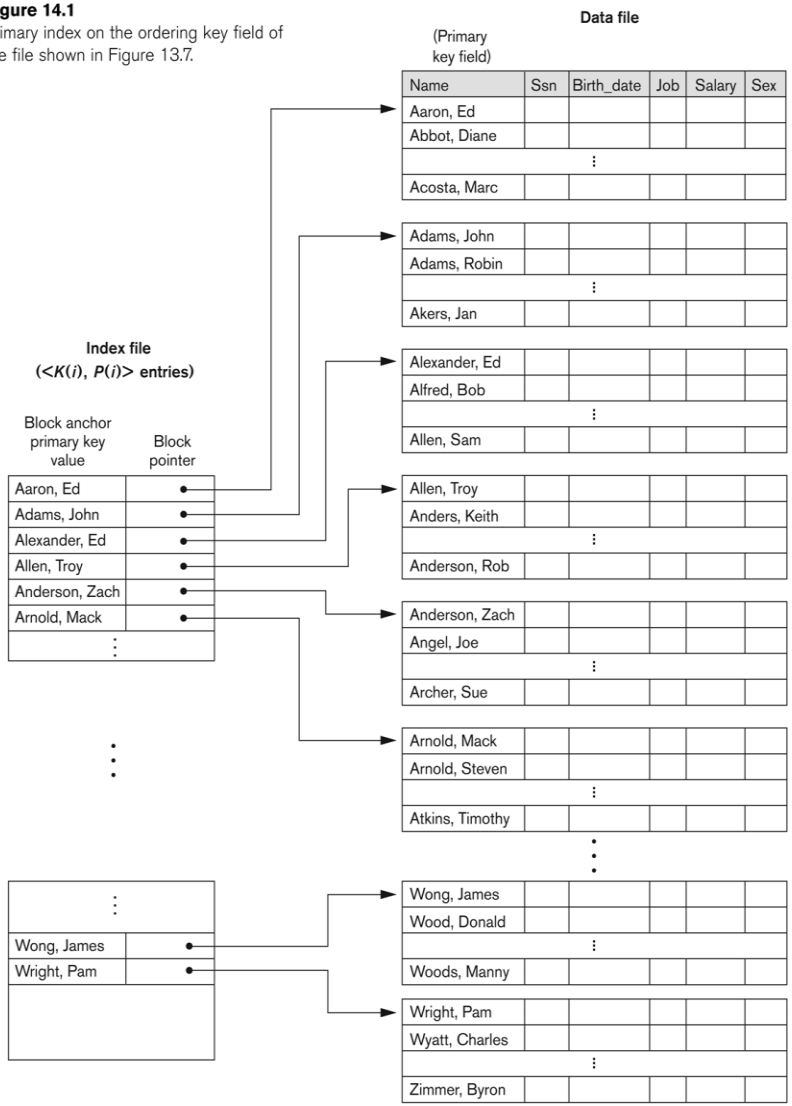
- Types of Single-level Ordered Indexes

  - Primary Indexes
  - Clustering Indexes
  - Secondary Indexes

- Multilevel Indexes

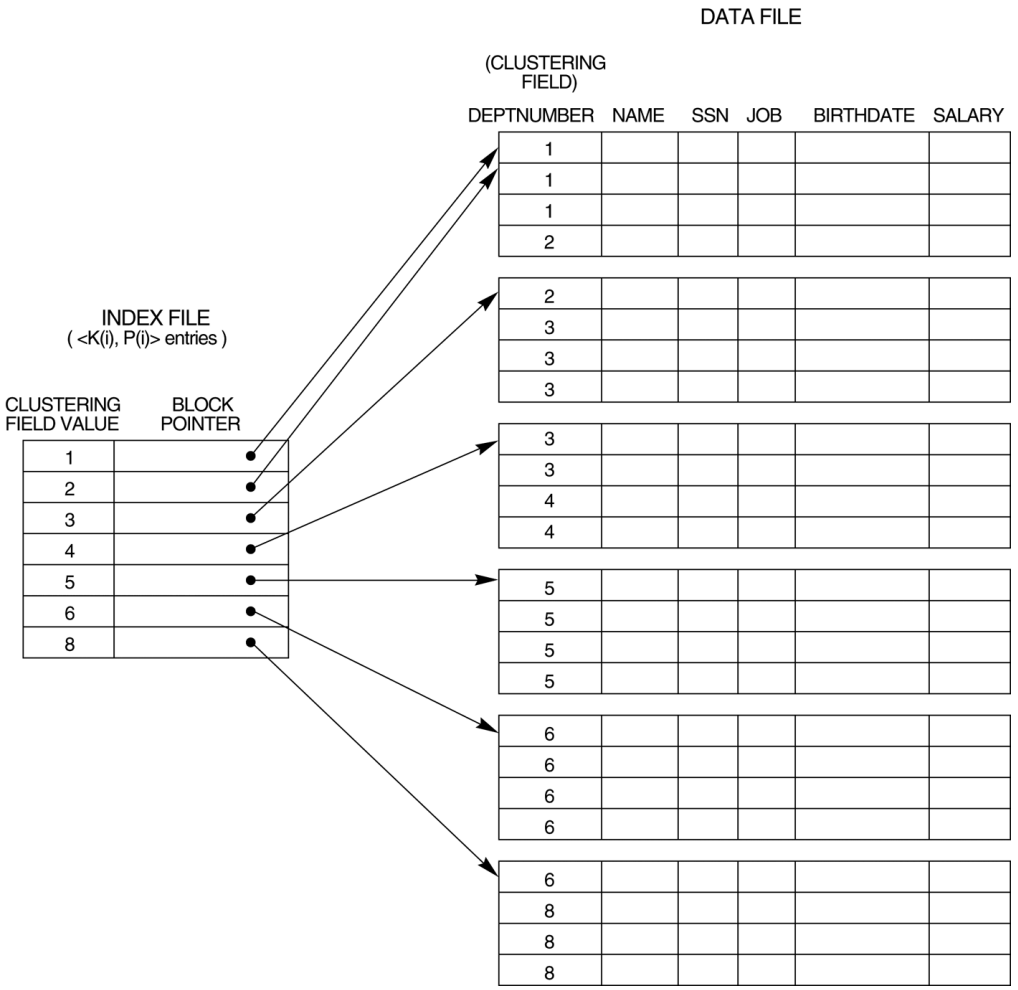
Primary index on the ordering key field

Figure 14.1
Primary index on the ordering key field of the file shown in Figure 13.7.



A Clustering Index Example

- A clustering index on the DEPTNUMBER ordering non-key field of an EMPLOYEE file.



Another Clustering Index Example

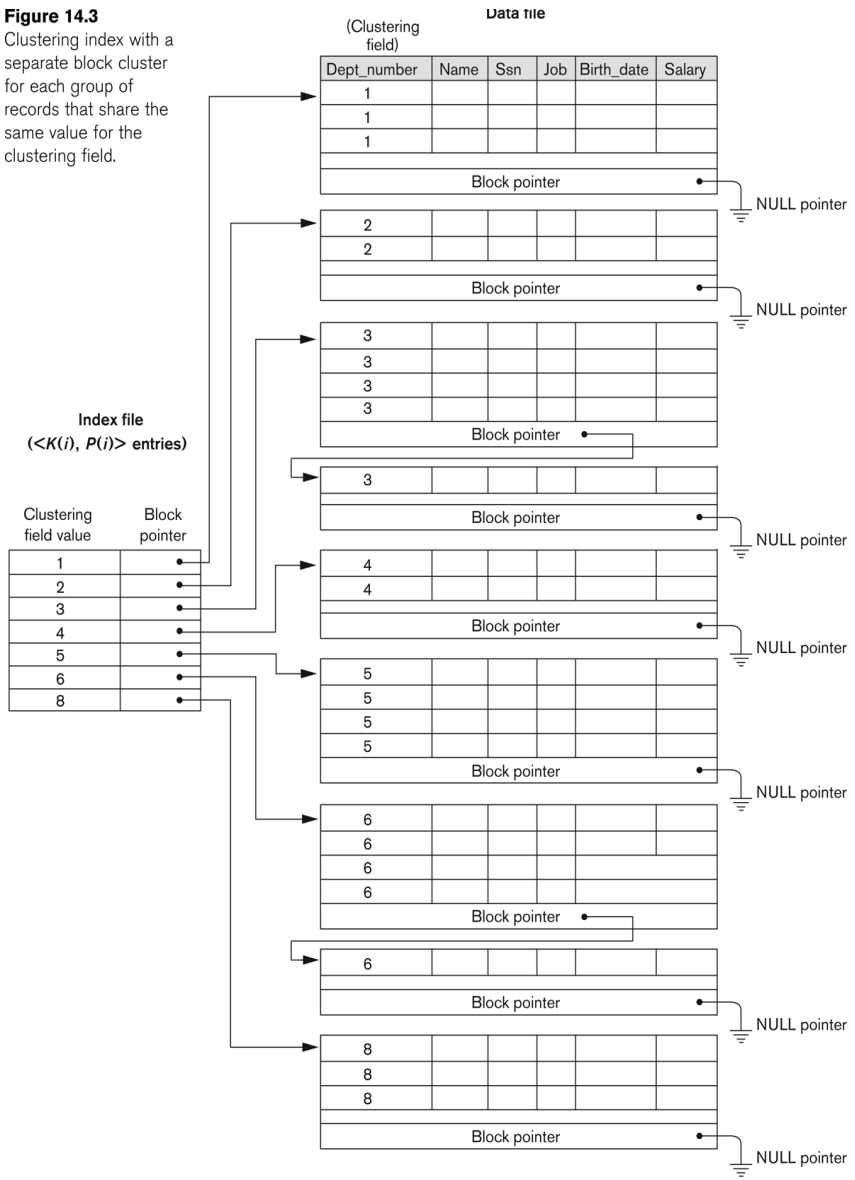
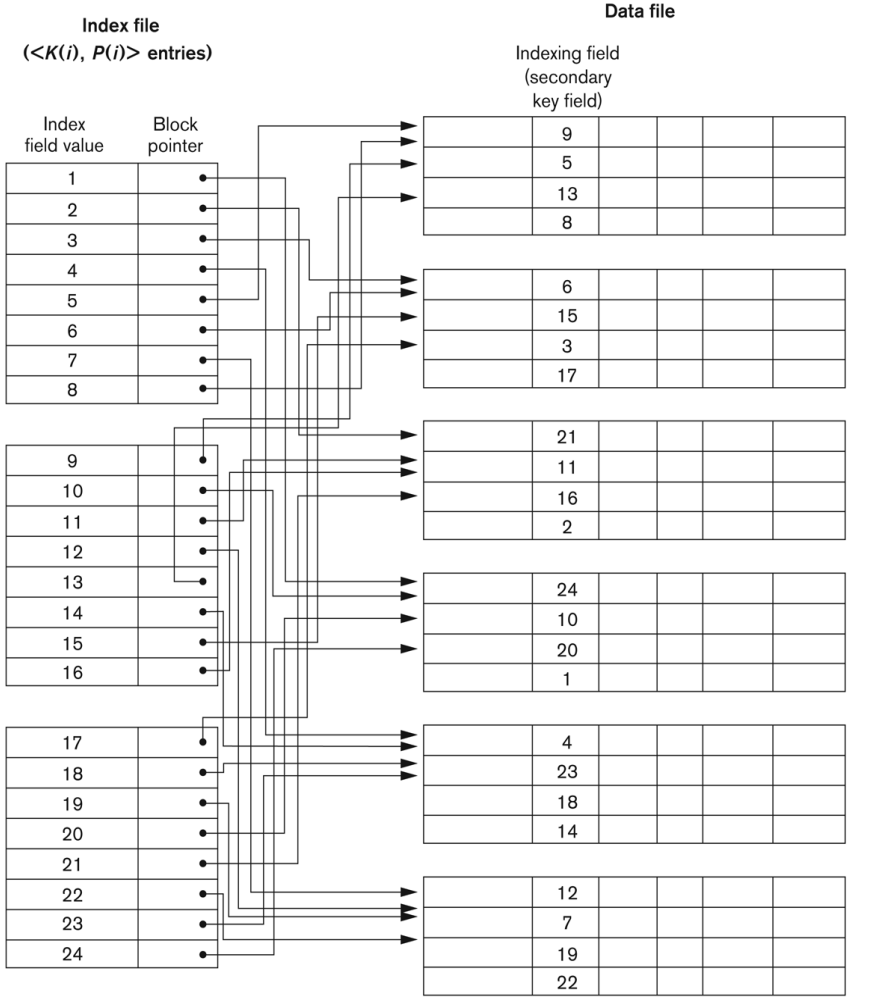


Figure 14.3
Clustering index with a separate block cluster for each group of records that share the same value for the clustering field.

Example of a Dense Secondary Index

Figure 14.4
A dense secondary index (with block pointers) on a nonordering key field of a file.



An Example of a Secondary Index

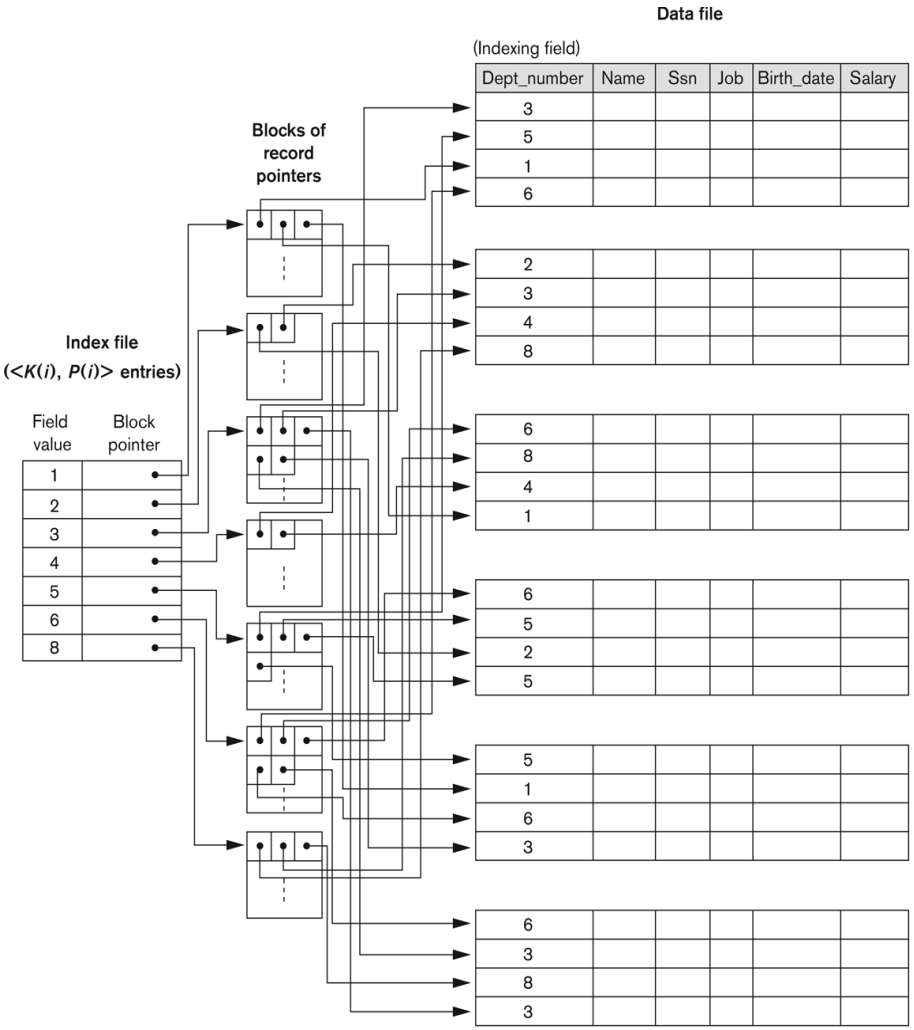


Figure 14.5
A secondary index (with record pointers) on a nonkey field implemented using one level of indirection so that index entries are of fixed length and have unique field values.

Properties of Index Types

TABLE 14.2 PROPERTIES OF INDEX TYPES

TYPE OF INDEX	NUMBER OF (FIRST-LEVEL) INDEX ENTRIES	DENSE OR NONDENSE	BLOCK ANCHORING ON THE DATA FILE
Primary	Number of blocks in data file	Nondense	Yes
Clustering	Number of distinct index field values	Nondense	Yes/no[a]
Secondary (key)	Number of records in data file	Dense	No
Secondary (nonkey)	Number of records[b] or Number of distinct index field values[c]	Dense or Nondense	No

[a]Yes if every distinct value of the ordering field starts a new block; no otherwise.

[b]For option 1.

[c]For options 2 and 3.