Indexing Methods

Indexes

- Index Files
  - Secondary or auxiliary files that help speed up data access in primary files
- Single level index
  - index file maps directly to the block or the address of the record
- Multi-level index
  - multiple levels of indirection among indexes

Definitions

- Indexing field (indexing attribute): The field on which an index structure is built (searching is fast on this field)

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.

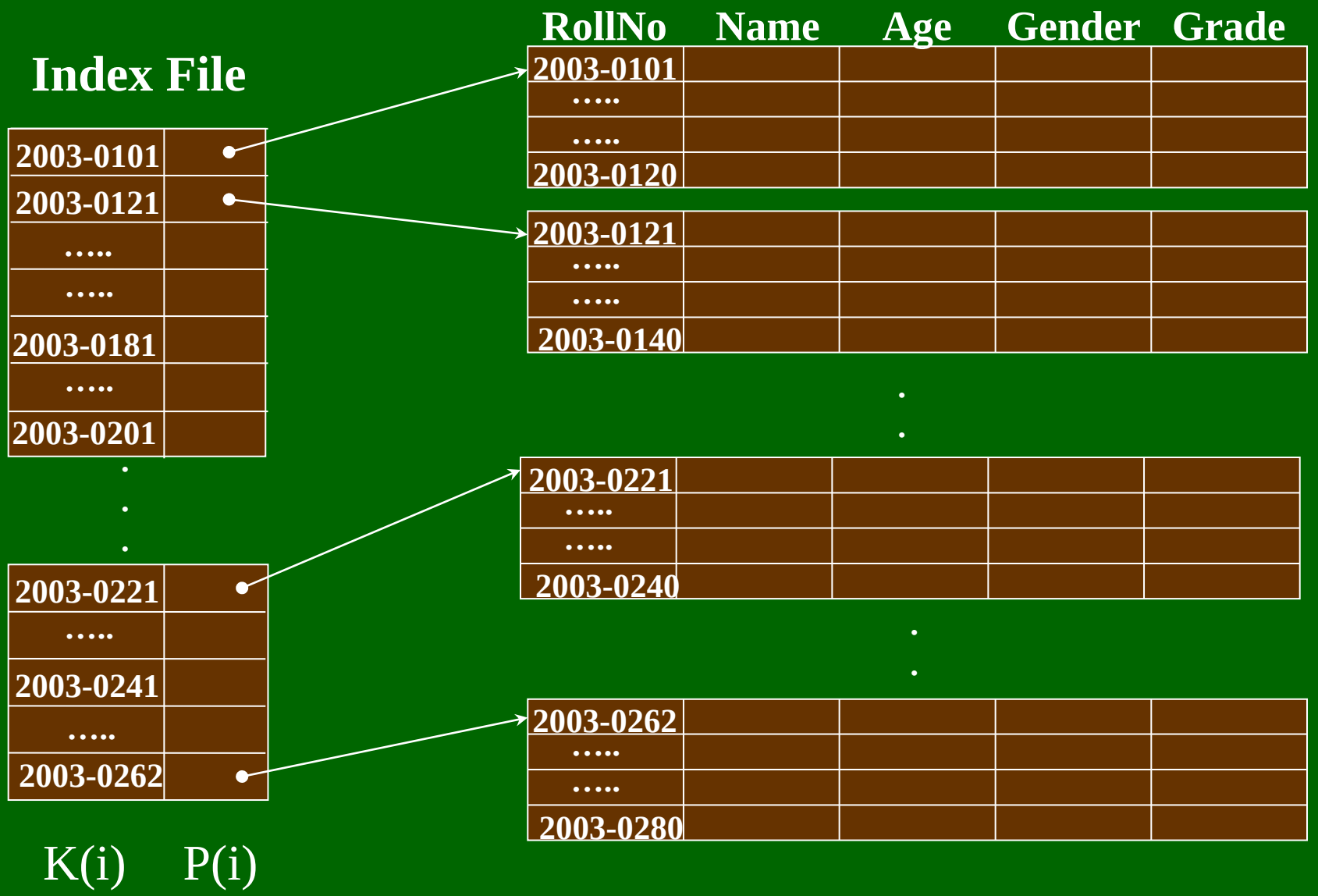
Indexes as Access Paths

- The index is called an Access Path on the field.
- The index file usually occupies less disk blocks than the data file because its entries are much smaller.
- A binary search on the index yields a pointer to a file record.

Primary Index

- Defined on an ordered data file
- Data file is ordered on a key-field
- Includes one index for each block in the data file; the index entry has a key field value for the first record in the block, which is called the block anchor.

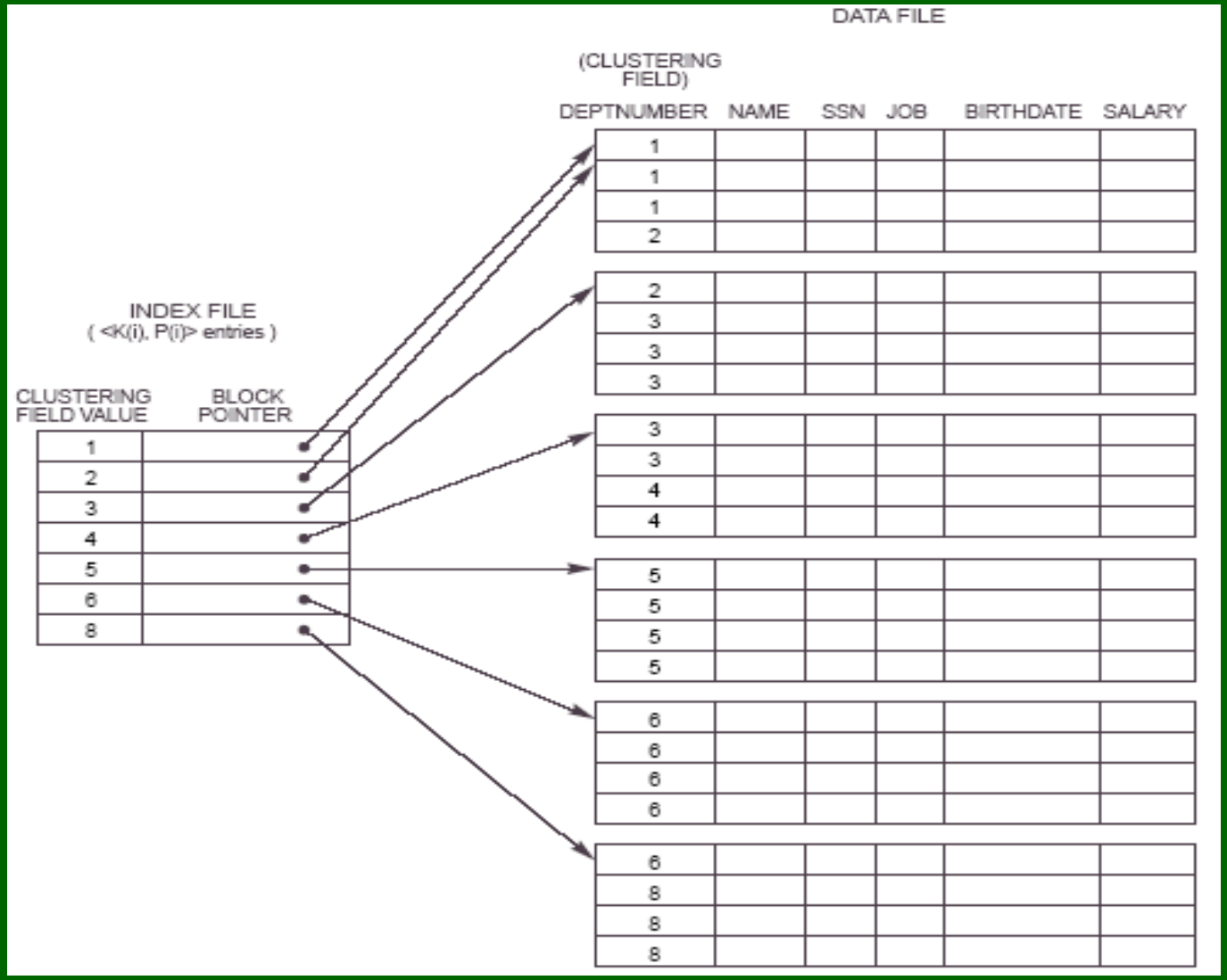
Primary Index



Clustering Index

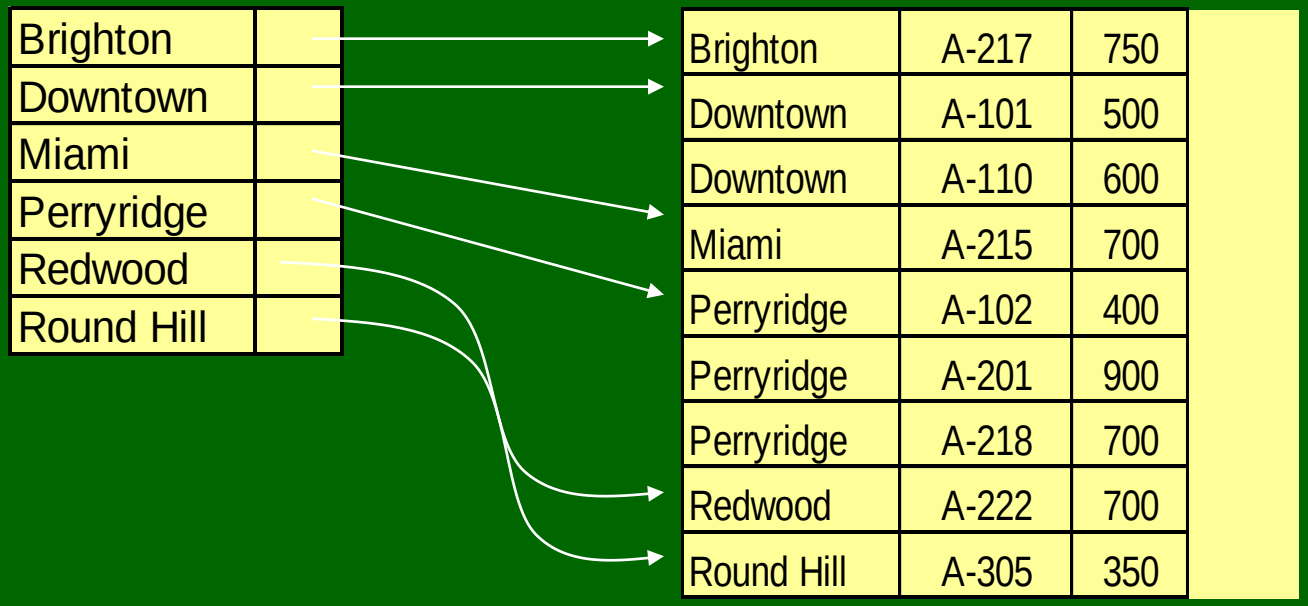
- Defined on an ordered data file
- Data file is ordered on a non-key-field
- Includes one index entry for each distinct value of the field; the index points to the first data block that contains records with that field value.

Clustering Index



Dense Index Files

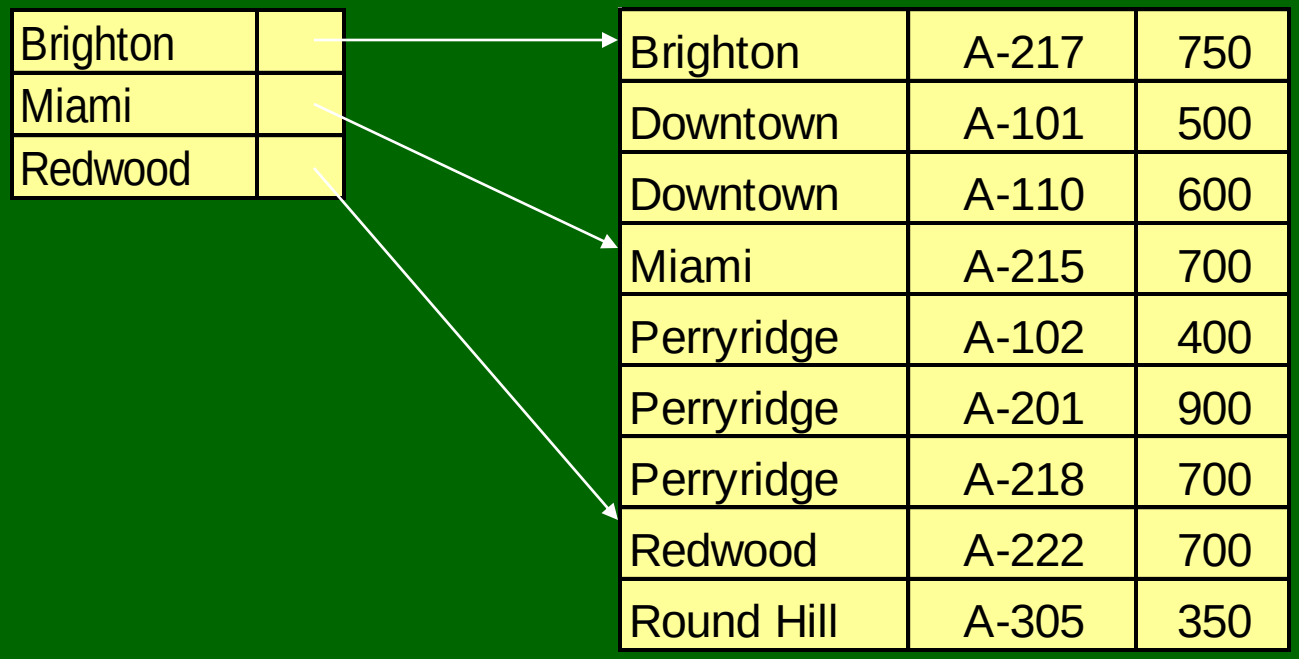
- Dense index – index record appears for every search-key value in the file.



Sparse Index Files

- A clustering index may be sparse.
- Index records for only some search-key values.
- To locate a record with search-key value *k* we:
  - Find index record with largest search-key value < *k*
  - Search file sequentially starting at the record to which the index record points
- Less space and less maintenance overhead for insertions and deletions.
- Generally slower than dense index for locating records.

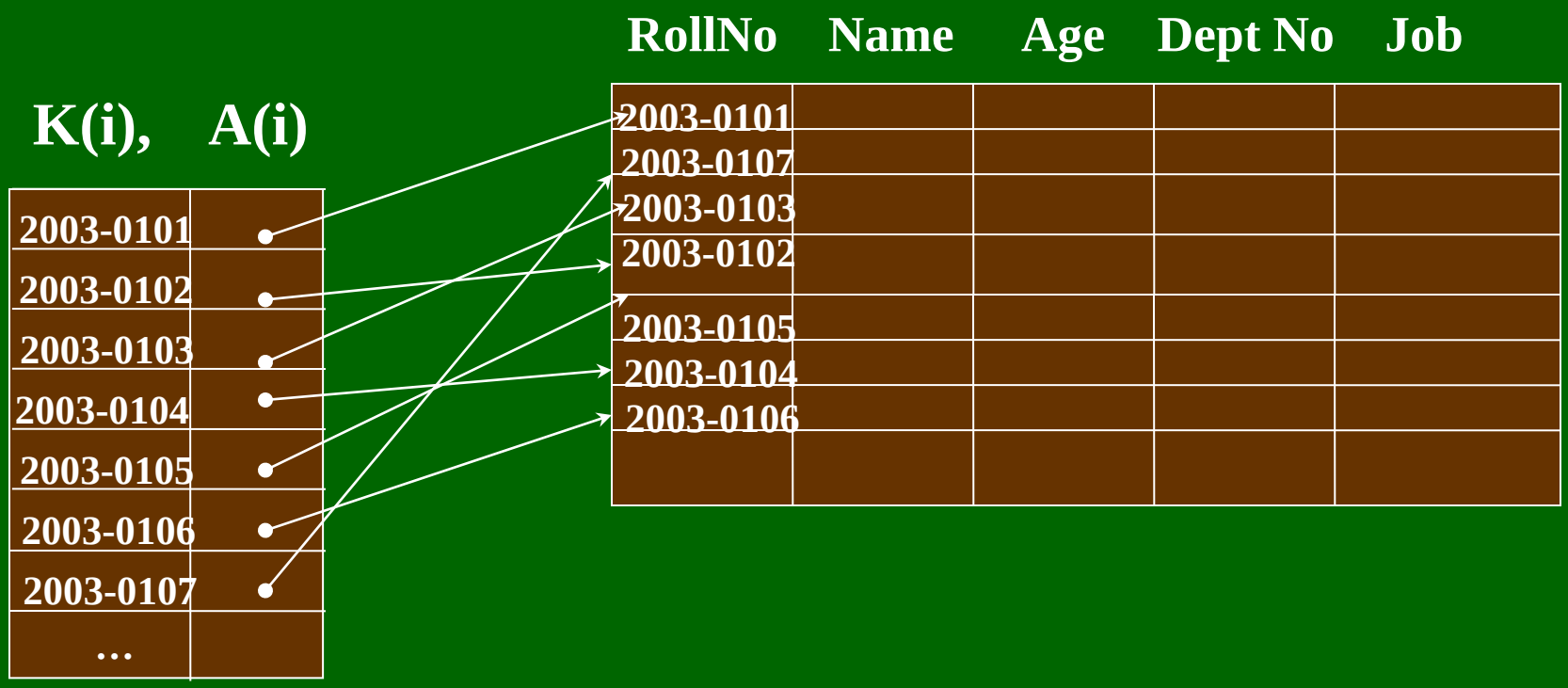
Sparse Index Files



Secondary Index

- Defined on an unordered data file
- Can be defined on a key field or a non-key field
- Includes one entry for each record in the data file

Secondary Index on Key Field



Has as many index entries as the number of records…

Secondary Index on Key Field

- Since key fields are unique, number of index entries equal to number of records

Secondary Index on non-key Field

- When a non-key field is indexed, duplicate values have to be handled.
- There are three different techniques for handling duplicates:
  - Duplicate index entries
  - Variable length records
  - Extra redirection levels

Duplicate Index Entries

K(i)	A(i)
2003-0101	
2003-0102	
2003-0102	
2003-0102	
2003-0102	
2003-0103	
2003-0103	
…	

Index entries are repeated for each duplicate occurrence of the non-key attribute.

Variable Length Records

- Use variable length records for index table in order to accommodate duplicate key entries
- For a given key K(i), there is a *set* of address pointers instead of a single address pointer

Extra Redirection Levels