Database Development and Design (CPT201)

Lecture 3a:
Indexing Techniques

Dr. Wei Wang

Department of Computing

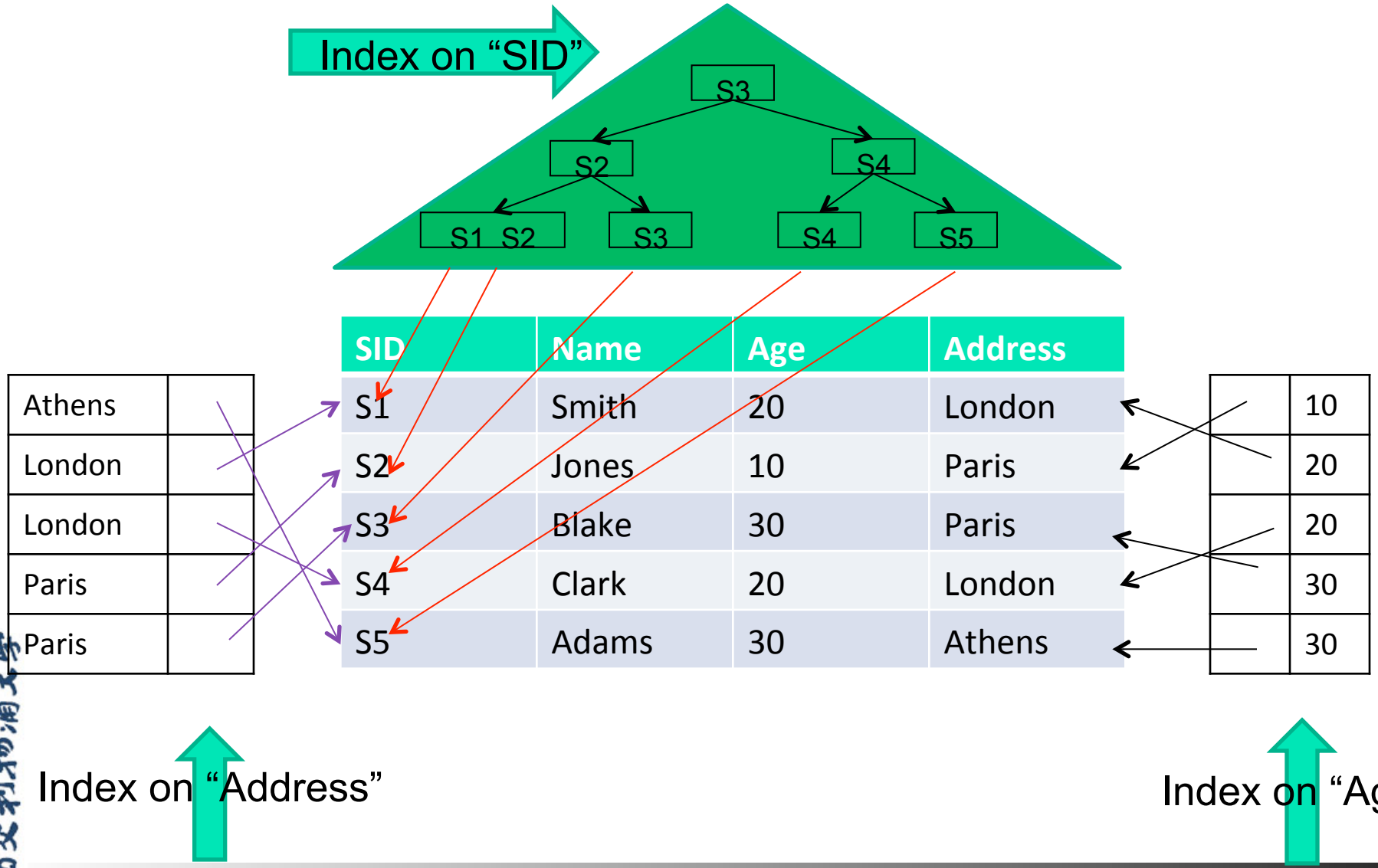
Learning Outcomes

- The Structure of Index
- Ordered Index
- Primary Index vs. Secondary Index
- Dense Index vs. Sparse Index
- Multilevel index

Motivation: Search Records

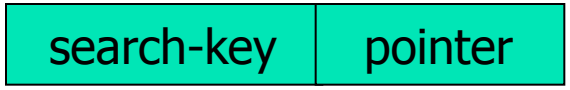
- To scatter records of a relation to different blocks is not efficient.
  - SELECT * FROM C;
    - problem: search all the blocks on the disk
    - solution: keep records of a certain relation on adjacent cylinders
  - SELECT * FROM C WHERE age=10;
    - problem: search all the blocks and check the condition on the disk
    - solution: create indices on some attributes
- Indexing mechanisms used to speed up access to the desired data.

Index


Index on "SID"
S3
S2
S4
S1 S2
S3
S4
S5
SID
Name
Age
Address
S1
Smith
20
London
S2
Jones
10
Paris
S3
Blake
30
Paris
S4
Clark
20
London
S5
Adams
30
Athens
Athens
London
London
Paris
Paris
10
20
20
30
30
Index on "Address"
Index on "Age

The Structure of Index

- Data file: collection of blocks holding records on disk
- Index file: an data structure allowing the DBMS to find particular records in a data file more efficiently.
  - An index file consists of records (called index entries) of the form:



- Search Key: one or set of attributes used to look up records in a file.
- Relationship: a search key K in the index file is associated with a pointer to a data-file record that has search key K.

Index Evaluation Metrics

- Access types (supported)
  - records with a specified value in the attribute or
  - records with an attribute value falling in a specified range.
- Access time
- Insertion time
- Deletion time
- Space overhead

Indexing Techniques

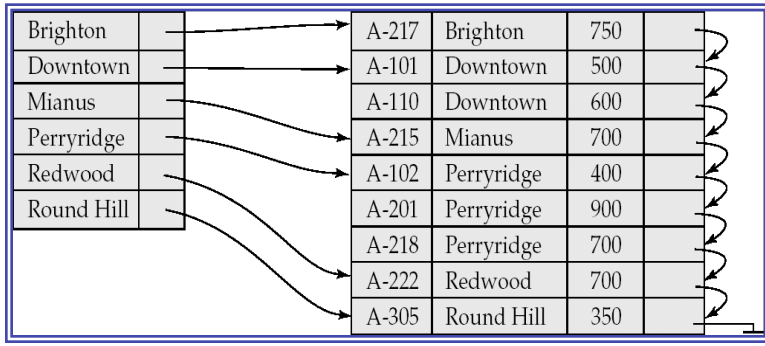
- Depending on the organisation of index file, an index can be:
  - an ordered Index where index entries are sorted on the search key value.
  - a hashing Index where hashing technique is employed to organise index entries.

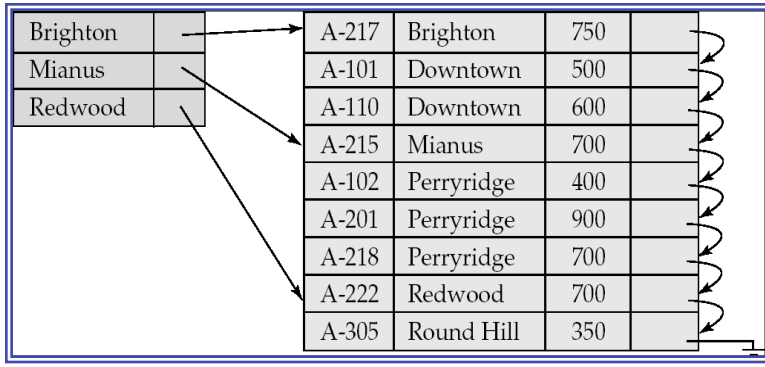
Ordered Indices

- Ordered index: index entries in the index are sorted on the search key value.
- An ordered index can be:
  - Dense index: index record appears for every search-key value in the file.
  - Sparse Index: contains index records for only some search-key values.

Dense Index vs. Sparse Index

- Index size
  - Sparse index is smaller
- Requirement on data file
  - The data file must be sequential file
- Lookup
  - Sparse index is smaller and may fit in memory
  - Dense index can directly tell if a record exists.
- Update
  - Sparse index requires less space and maintenance for insertion and deletion.
- Good tradeoff: sparse index with an index entry for every block in file, corresponding to least search-key value in the block.

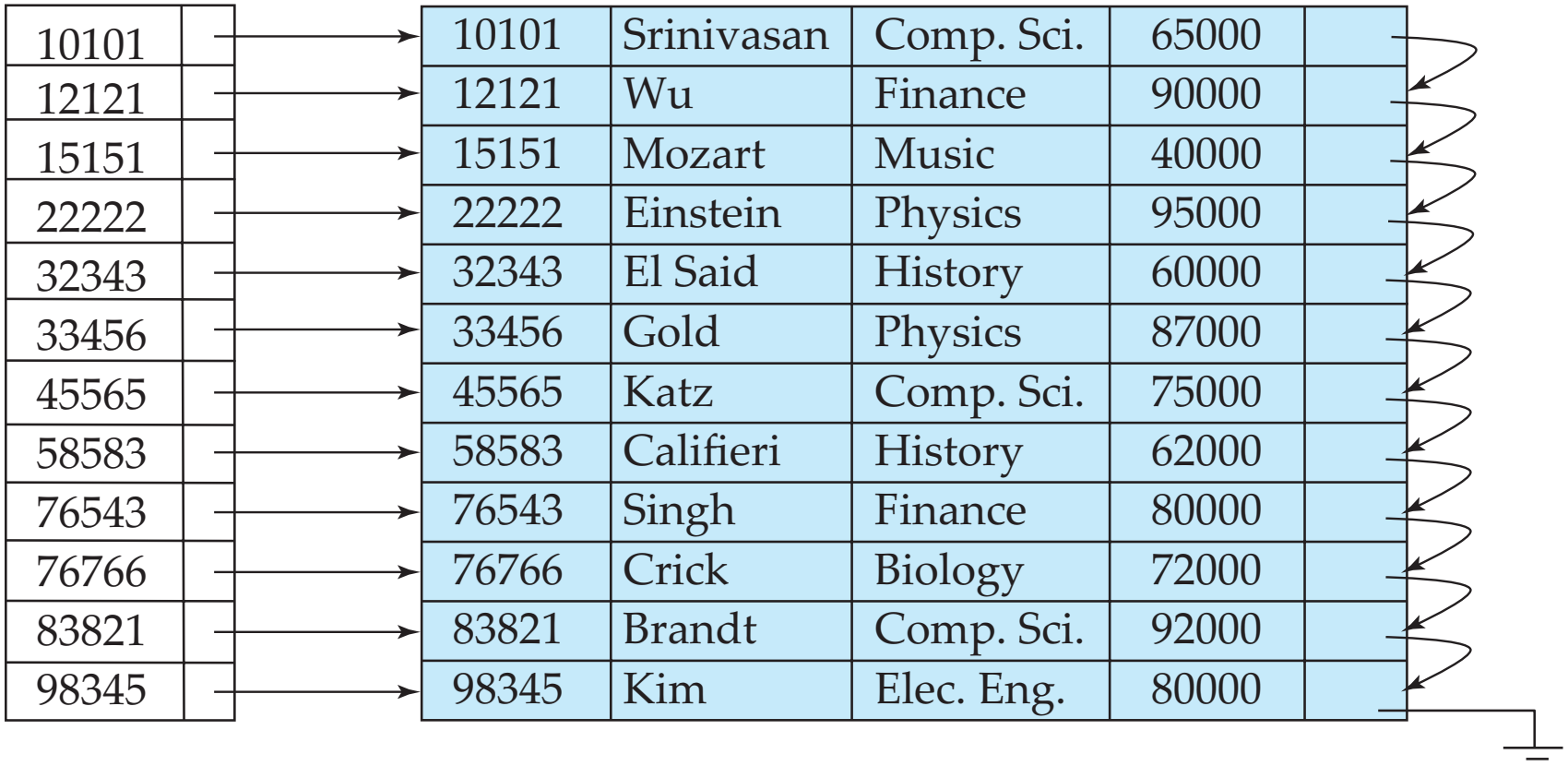




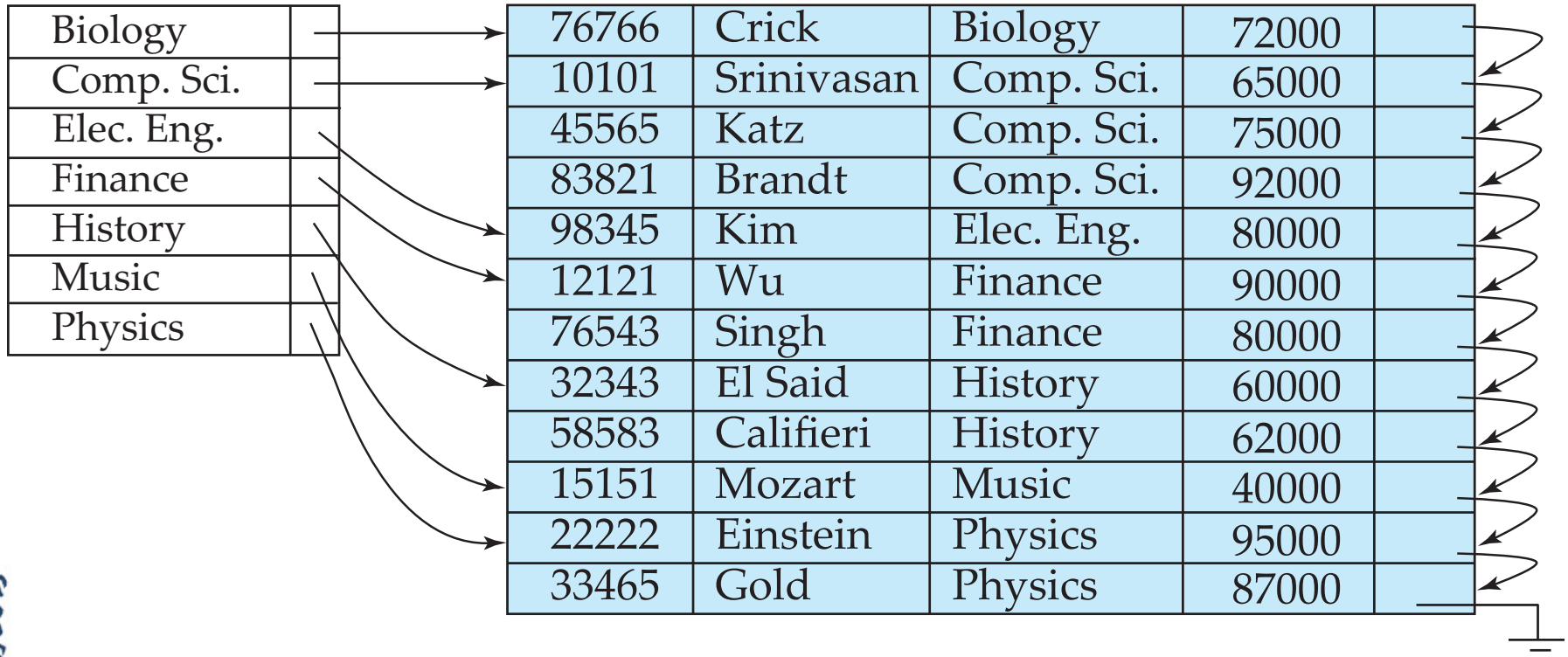
Ordered Indices cont'd

- An ordered index can also be:
  - Primary index: an index whose search key specifies the sequential order of the file.
    - Also called clustering index. The search key of a primary index is usually but not necessarily the primary key.
    - Can be sparse
  - Secondary index: an index whose search key specifies an order different from the sequential order of the file.
    - Also called non-clustering index.
    - Can not be sparse
- Index-sequential file: ordered sequential file with a primary index.

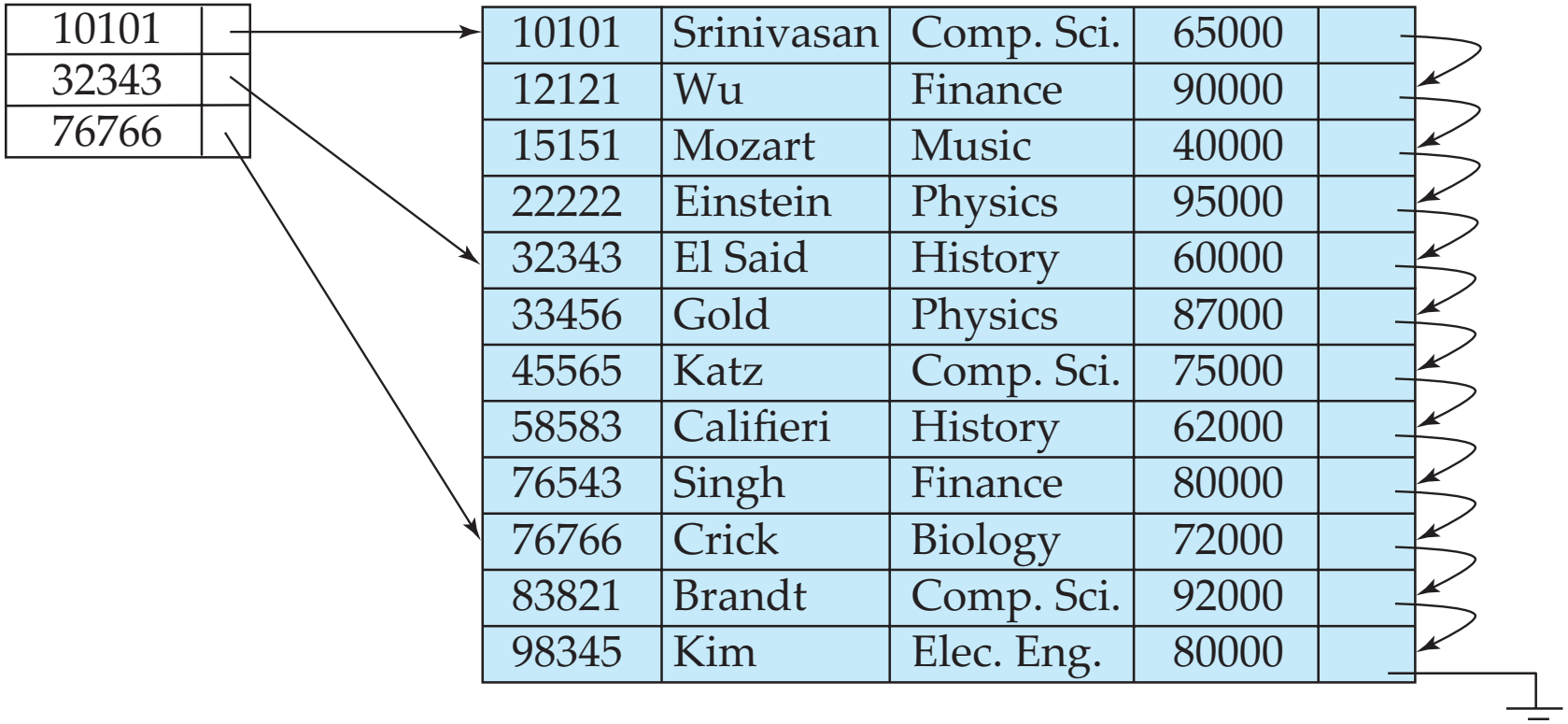
Dense Index Files



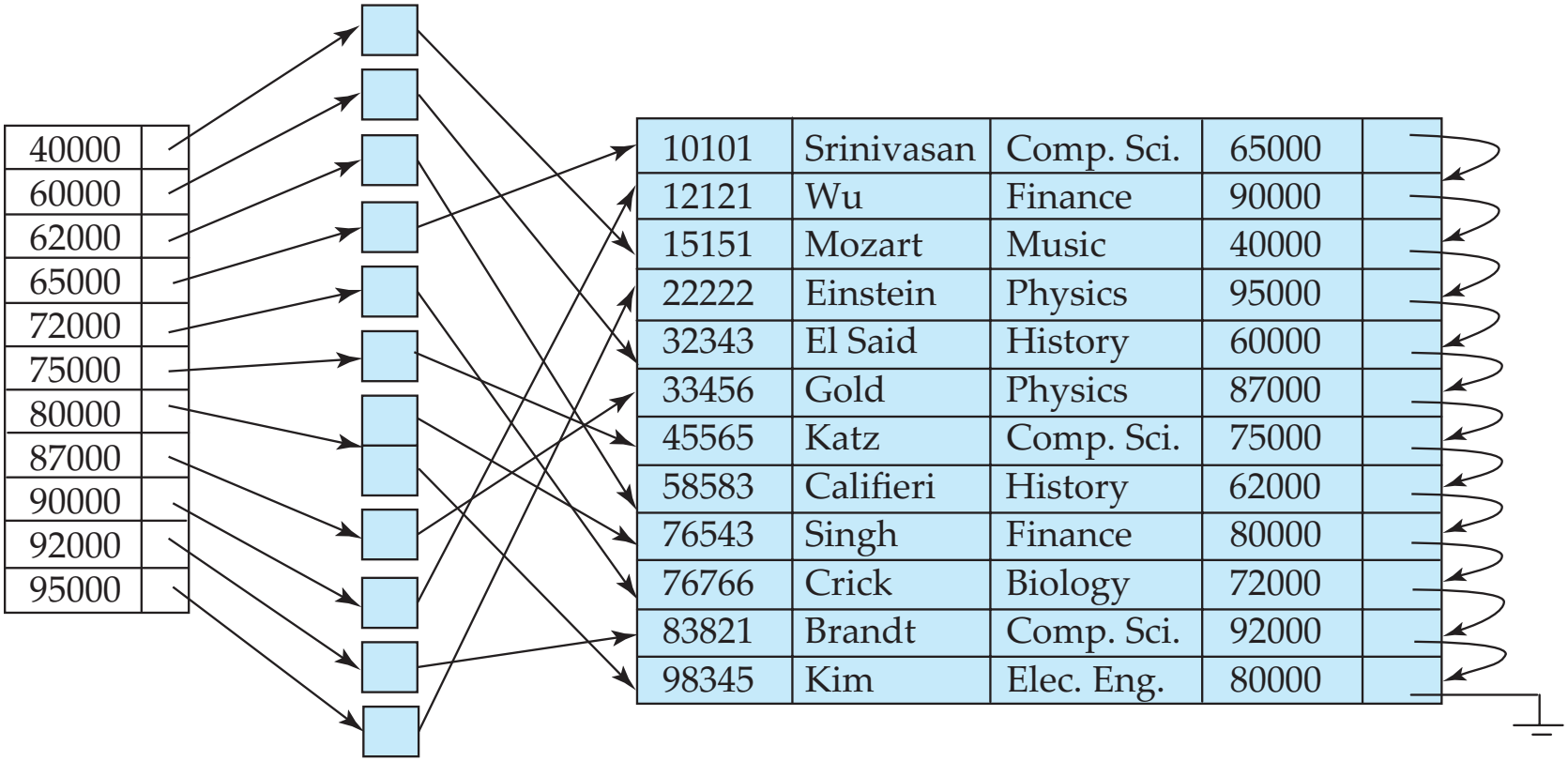
Dense Index Files cont'd



Sparse Index Files



Secondary Index



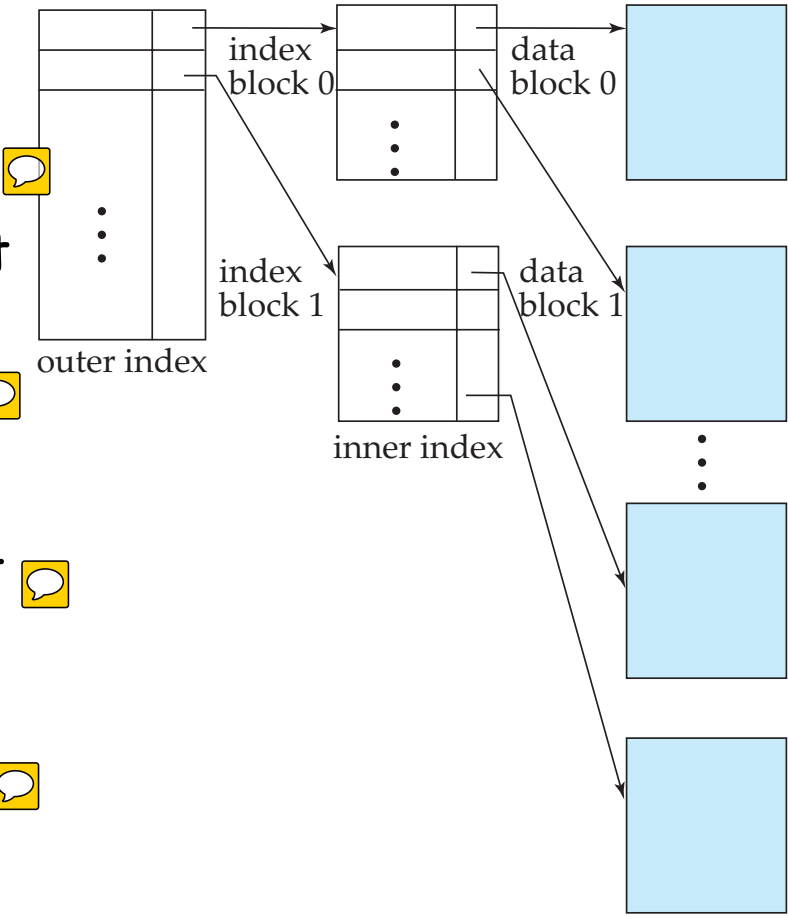
- Index record points to a bucket that contains pointers to all the actual records with that particular search-key value.
- Secondary indices have to be dense

Primary and Secondary Indices

- Indices offer substantial benefits when searching for records.
- But updating indices imposes overhead on database modification - when a file is modified, every index on the file must be updated
- Sequential scan using primary index is efficient
- But a sequential scan using a secondary index is expensive
  - Each record access may fetch a new block from disk
  - Block fetch requires about 5 to 10 milliseconds; versus about 100 nanoseconds for memory access

Multilevel Index (Index on index)

- If primary index does not fit in memory, access becomes expensive.
- Solution: treat primary index kept on disk as a sequential file and construct a sparse index on it.
  - outer index – a sparse index of primary index
  - inner index – the primary index file
- If even outer index is too large to fit in main memory, yet another level of index can be created, and so on.
- Indices at all levels must be updated on insertion or deletion from the file.



Index Definition in SQL

- Create an index
  - create index <index-name> on <relation-name> (<attribute-list>)
  - E.g.: create index b-index on branch(branch_name)
- To drop an index
  - drop index <index-name>
- Most database systems allow specification of type of index.

End of Lecture

- Summary
  - The Structure of Index
  - Ordered Indices
  - Primary index vs. Secondary index
  - Dense index vs. sparse index
  - Multilevel index
- Reading
  - Database System Concepts, 6th edition, chapter 11.1, 11.2
  - Database System Concepts, 7th edition, chapter 14.1, 14.2