SE212
Physical Database Design

Agenda

1. Final exam date changed (Monday 14th 1pm-4pm) (3hrs)
2. The Physical Database Design
3. Process

The Physical Database

- The primary goal of physical database design is data processing efficiency.
- Every decreasing costs for computer technology per unit of measure (both speed and space)

The Physical Database

- To minimize the time required by users to interact with the information system.
- Thus, we concentrate on how to make processing of physical files and databases efficient, with less attention on minimizing the use of space

The Physical Database

- Designing physical files and databases requires certain information that should have been collected and produced during systems development phases.

The information needed for physical file and database design includes

- Normalized relations, including estimates for the range of the number of rows in each table
- Definitions of each attribute, along with physical specifications such as maximum possible length
- Descriptions of where and when data are used in various ways (entered, retrieved, deleted, and updated)
- Expectations or requirements for response time and data security, backup, recovery, retention, and integrity
- Descriptions of the technologies (database management system) used for implementing the database

Designing Tables

- The first step in physical database design is to map the normalized relations shown in the logical design to tables.
- The importance of this step should be obvious because tables are the primary unit of storage in relational databases. However, if adequate work was put into the logical design, then translation to a physical design is that much easier.
- physical database components (tables, constraints, indexes, views, and so on).

Designing table

1. Each normalized relation becomes a table.

2. Each attribute within the normalized relation becomes a column in the corresponding table.

   - A unique column name with in the table
   - A data type, and for some data types, a length.
   - Whether column values are required or not. This takes the form of a NULL or NOT NULL clause for each column.

3. The unique identifier of the relation is defined as the primary key of the table.

4. Relationships among the normalized relations become referential constraints in the physical design.

Adding Indexes for Performance

- Indexes provide a fast and efficient means of finding data rows in tables, much like the index at the back of a book helps you in quickly finding specific references.
- Although the implementation in the database is more complicated than this, it’s easiest to visualize an index as a table with one column containing the key value and another containing a pointer to where the row with that key value physically resides in the table, in the form of a row ID or a relative block address (RBA).

Adding Indexes for Performance

- Indexes provide faster searches than scanning tables for two reasons.
- First, index entries are considerably shorter than typical table rows, so many more index entries fit per physical file block than the corresponding table rows. Therefore, when the database must scan the index sequentially looking for matching rows, it can get a lot more index entries with a single read to the file on disk than a corresponding read to the file holding the table.
- Second, index entries are always maintained in key sequence, which is not at all true of tables. The RDBMS software can take advantage of this by using binary search techniques that remarkably reduce search times and the resources required for searching.

The physical database design process

Has two stages:

- First-cut database design stage
- Optimized database design

First-cut database design stage

- To use the conceptual constructs of the logical-level schema of the target database management system to develop a design that matches the conceptual data models as closely as possible.
- Each entity type in the conceptual data model becomes a table, with each of the attributes of eth entity type becoming a column of that table.

First-cut database design stage

- Need to identified foreign keys and add foreign key columns of the table.
- Naming the tables and columns that relate the schema design back to the conceptual data model within the limitations of database management systems place on the lengths of names.
- Example → EMPLOYEE QUALIFICATION entity would be called employee_qualification or emplyeeQualification to cope the restriction that spaces are not allowed in SQL names.

First-cut database design stage

- Column names must be unique within a table.
- To name foreign key columns with the same names as the column that they correspond to in the table that is referenced by the foreign key.
- Each column is defined with a datatype.

First-cut database design stage

- Another important element of the first-cut database design is the specification of the physical file storage for the database.
- The database management systems manage the actual storage of data but most provide mechanisms for the designer to control the allocation of tables to particular physical structures.
- These physical structure called tablespace, file groups or some other name.
- The allocation of tables to tablespaces is determined principally by the likely volumes of data for each tables and how data is to be collected from different tables.

Optimized database design stage

- First-cut design gives a database design that closely the conceptual data model.
- However, within the stared requirements for any information system are a number of non-functional requirements that specify performance targets for the overall system such as the maximum time a user must wait after submitting a request (a query on the database).

The two main strategies

The two main strategies for improving performance of a database are to:

- Make use of the built-in facilities of the database management system
- Compromise on the design of the logical schema

The two main strategies

- The most database management systems are the ability to cluster data and the ability to create indexes.
- Data clustering means arranging data on the disk that related data is placed as closely together as possible.
- An index provides an alternative way to access data other than searching through all the physical records associated with a particular logical table.
- It enables the database management system to know where to go to access any particular piece of data.

The two main strategies

- An index may built on a single column or multiple columns from the same table.
- Using an index improves retrieval performance by reducing the number of disk accesses required to query the data.

Weekly Worksheet
→ File Organizations

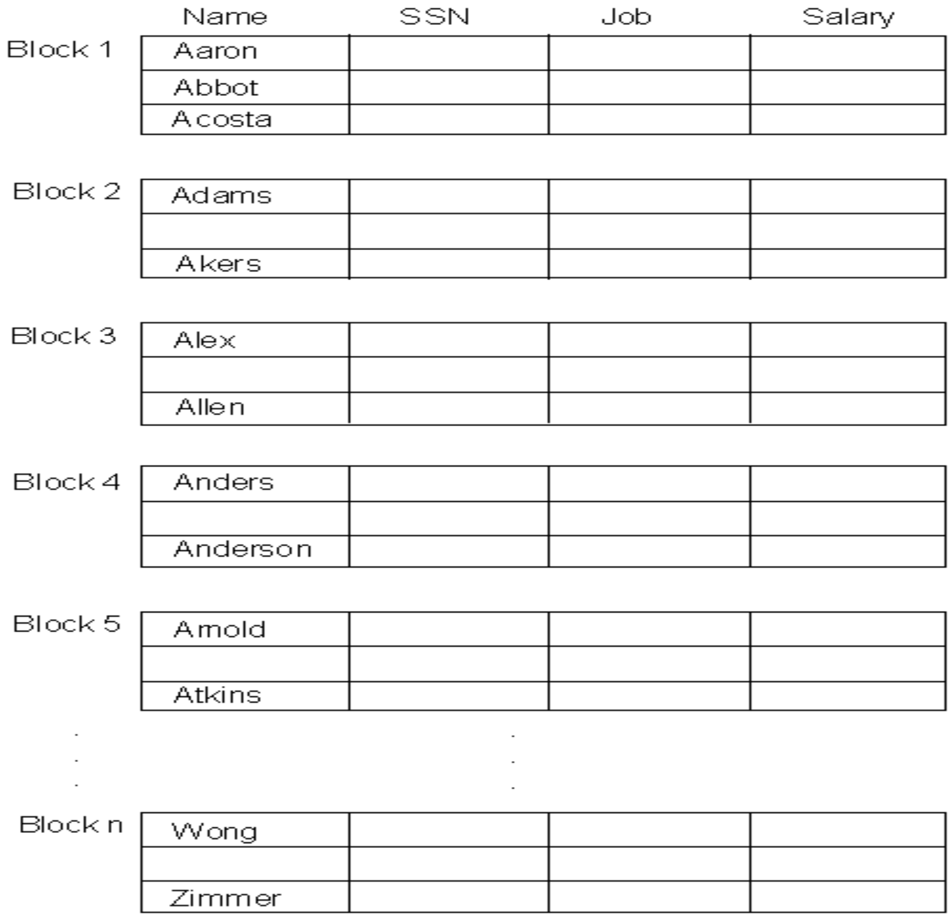
Search and present brief description of

- Sequential file organizations
- Indexed file organizations
- Hashed

Sequential file organization

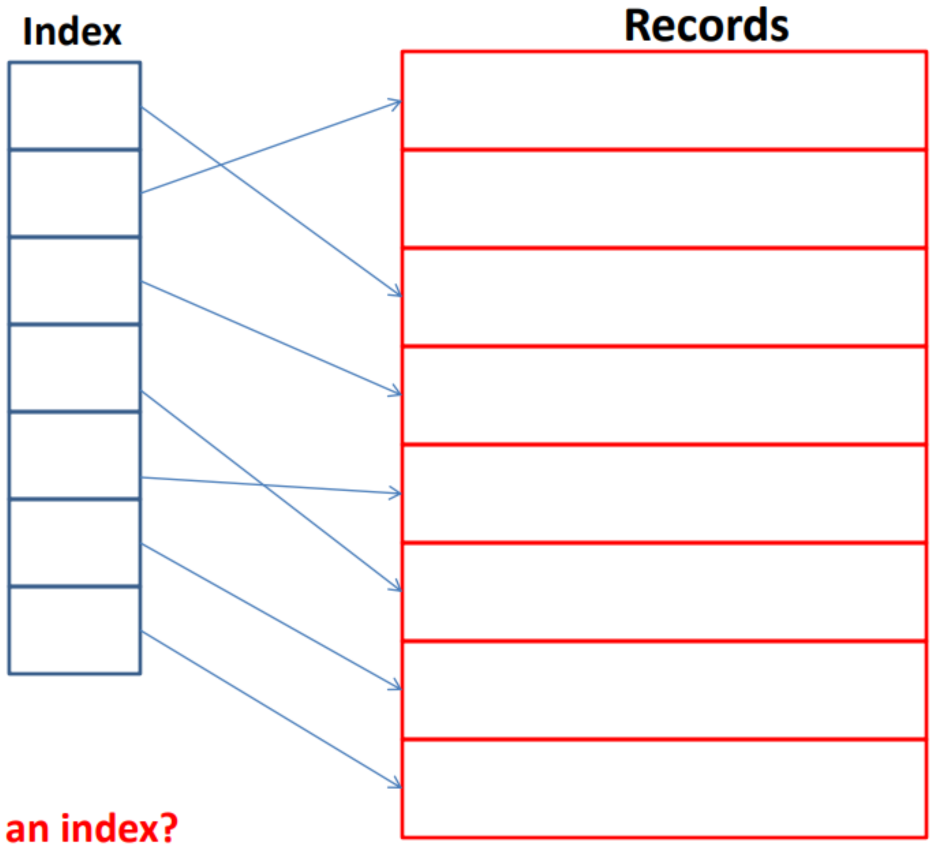
- Records are stored and accessed in a particular order sorted using a key field.
- The order of the records is fixed
- Because the record in a file are sorted in a particular order, better file searching methods like the binary search technique can be used to reduce the time used for searching a file .

Sequential file organization



	Name	SSN	Job	Salary
Block 1	Aaron			
	Abbot			
	Acosta			
Block 2	Adams			
	Akers			
Block 3	Alex			
	Allen			
Block 4	Anders			
	Anderson			
Block 5	Arnold			
	Atkins			
. . .		. . .		
Block n	Wong			
	Zimmer			

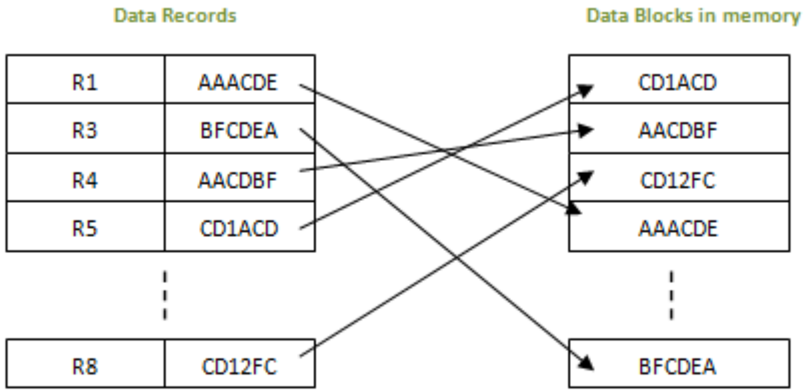
Index file organization



What is an index?

Indexed file organizations

- Advanced than the sequential file organization method
- For each primary key, an index value is generated and mapped with the record.
- The index is nothing but the address of record in the file.



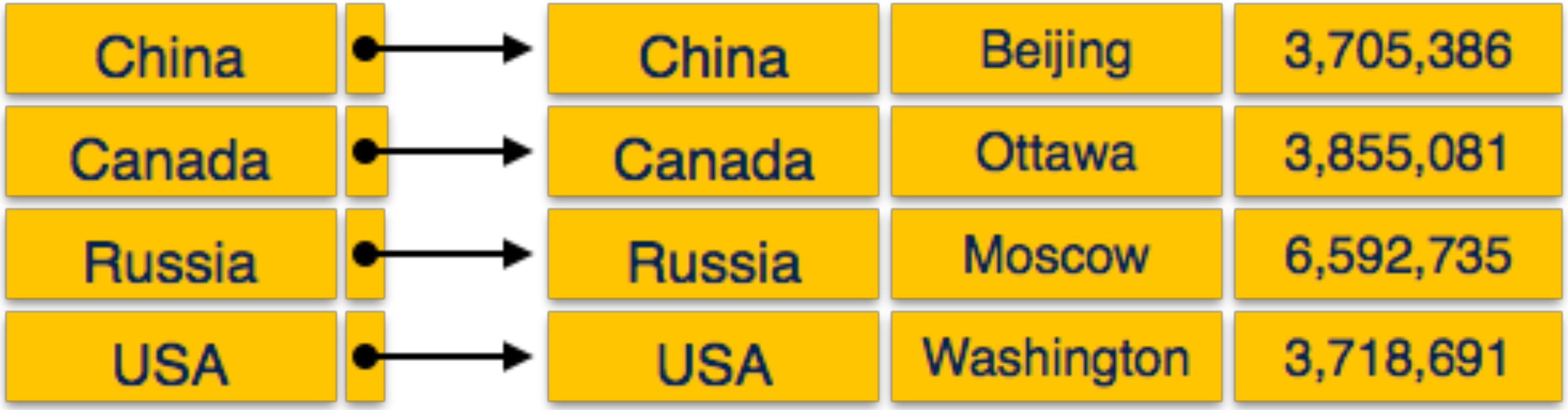
Data Records

R1	AAACDE
R3	BFCDEA
R4	AACDBF
R5	CD1ACD
R8	CD12FC

Data Blocks in memory

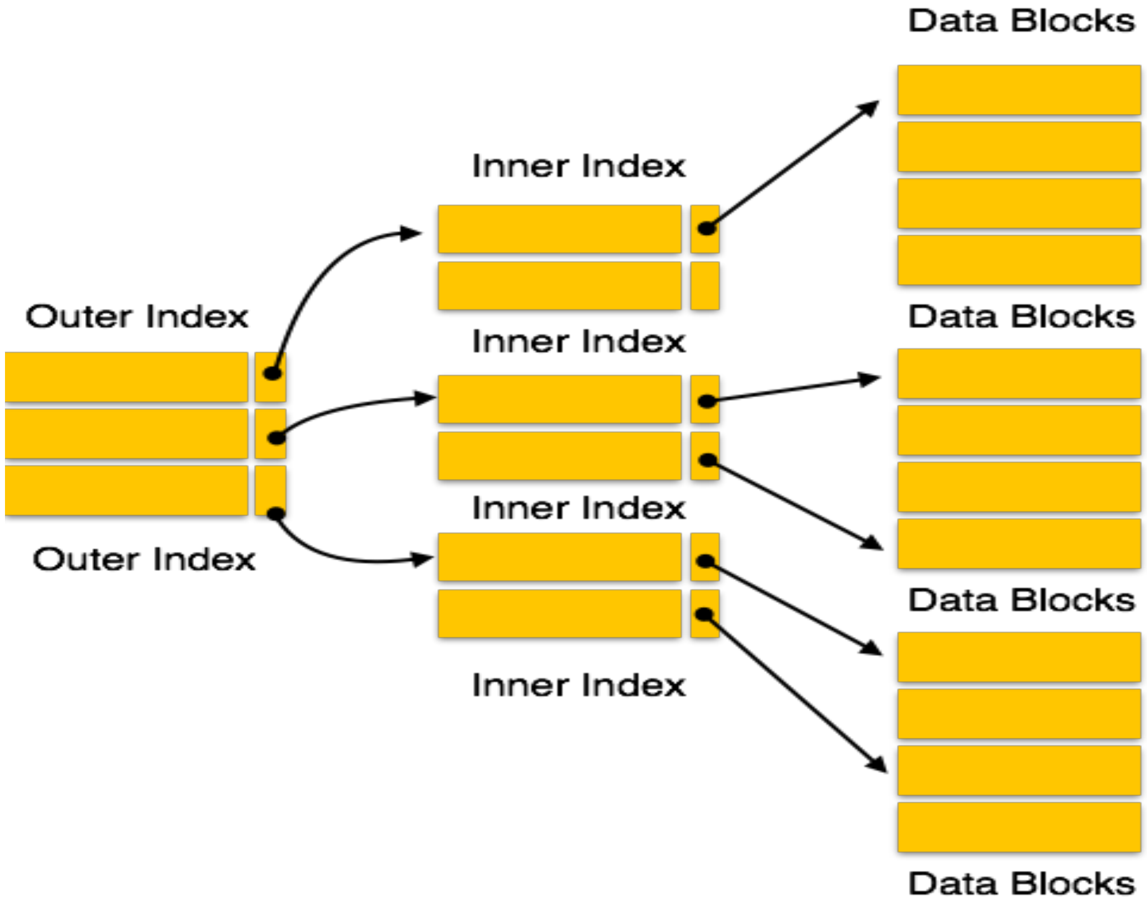
CD1ACD
AACDBF
CD12FC
AAACDE
BFCDEA

Dense Index



China	China	Beijing	3,705,386
Canada	Canada	Ottawa	3,855,081
Russia	Russia	Moscow	6,592,735
USA	USA	Washington	3,718,691

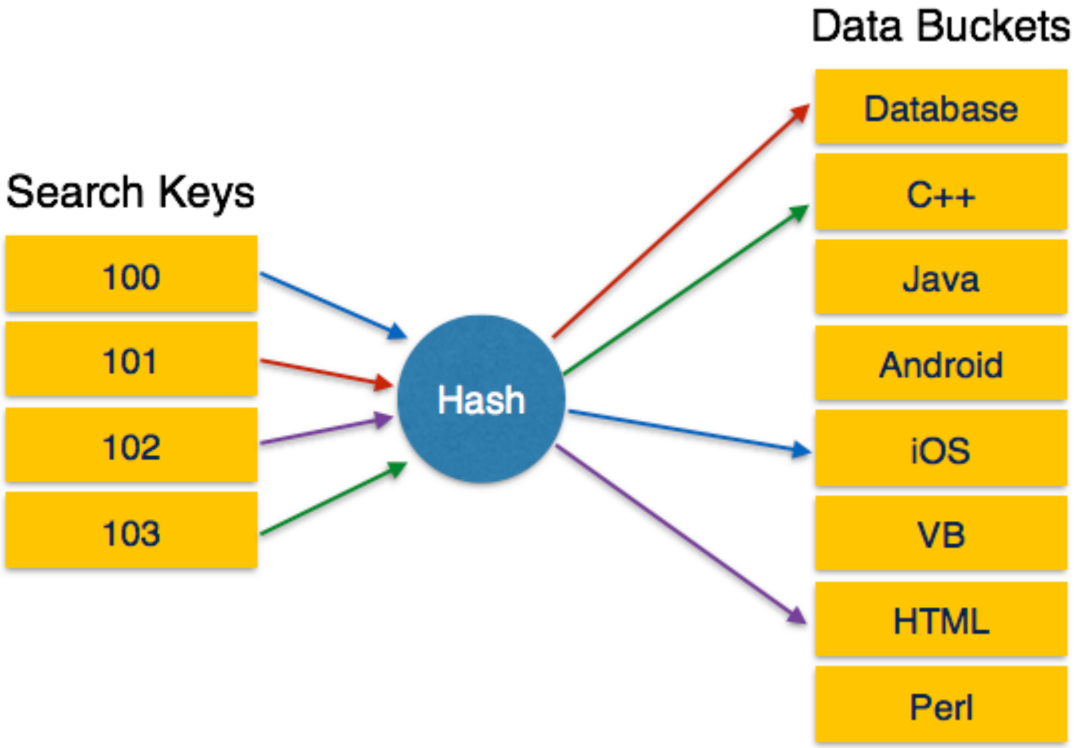
Multi-level index



Hash organization

- Bucket – a hash file stores data in bucket format. Bucket is considered a unit of storage. A bucket typically stores one complete disk block, which in turn can store one or more records.
- Has function - A hash function, **h,** is a mapping function that maps all the set of search-keys **K** to the address where actual records are placed. It is a function from search keys to bucket addresses.

Hash organization



Reference

- Gordon, K. (2013). Principles of Data Management: Facilitating information sharing. Second edition. BCS. United Kingdom.