



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : I - Introduction

Date of Lecture: 04.08.2020

Topic of Lecture: File systems versus Database systems

Introduction :

- File system is a software that manages and organizes the files in a storage medium within a computer. DBMS is a software for managing the database.

Prerequisite knowledge for Complete understanding and learning of Topic:

Data Structures, Program, Table, Tuples, Files, Records.

Detailed content of the Lecture:

File System:

- A file system is a technique of arranging the files in a storage medium like a hard disk, pen drive, DVD, etc.
- It helps you to organize the data and allows easy retrieval of files when they are required.
- It mostly consists of different types of files like mp3, mp4, txt, doc, etc. that are grouped into directories.
- A file system enables you to handle the way of reading and writing data to the storage medium.
- It is directly installed into the computer with the Operating systems such as Windows and Linux.

Features of a File system:

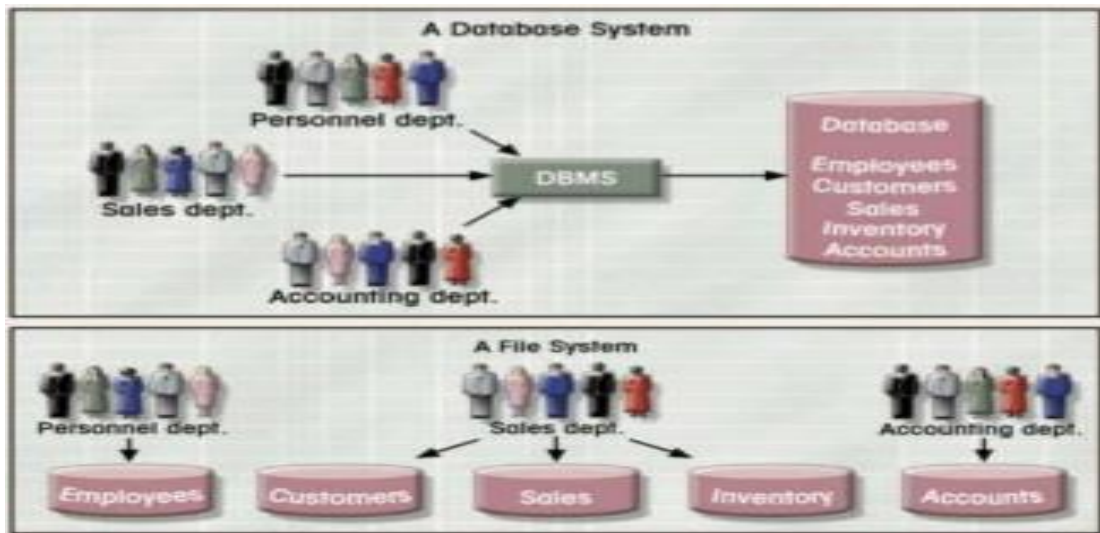
- It helps you to store data in a group of files.
- Files data are dependent on each other.
- C/C++ and COBOL languages were used to design the files.
- Shared File System Support
- Fast File System Recovery.

DBMS:

- Database Management System (DBMS) is a software for storing and retrieving user's data while considering appropriate security measures.
- It consists of a group of programs that manipulate the database. The DBMS accepts the request for data from an application and instructs the DBMS engine to provide the specific data. In large systems, a DBMS helps users and other third-party software to store and retrieve data.

Features of DBMS:

- A user-accessible catalog of data
- Transaction support
- Concurrency control with Recovery services
- Authorization services
- The value of data is the same at all places.
- Offers support for data communication
- Independent utility services
- Allows multiple users to share a file at the same time



Disadvantages:

- Data Redundancy
- Data Inconsistency
- Difficulty in Accessing Data
- Data Isolation
- Integrity Problem
- Atomicity Problem
- Concurrent Access Anomalies
- Security Problem

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=8dYCZFTtYjY>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 36 - 40)

Course Faculty

Verified by HOD



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs.R.Pavithra

Unit : I - Introduction

Date of Lecture: 05.08.2020

Topic of Lecture: Characteristics of database

Introduction :

- Provides security and removes redundancy.
- Self-describing nature of a database system.
- Insulation between programs and data abstraction. Support of multiple views of the data.

Prerequisite knowledge for Complete understanding and learning of Topic:

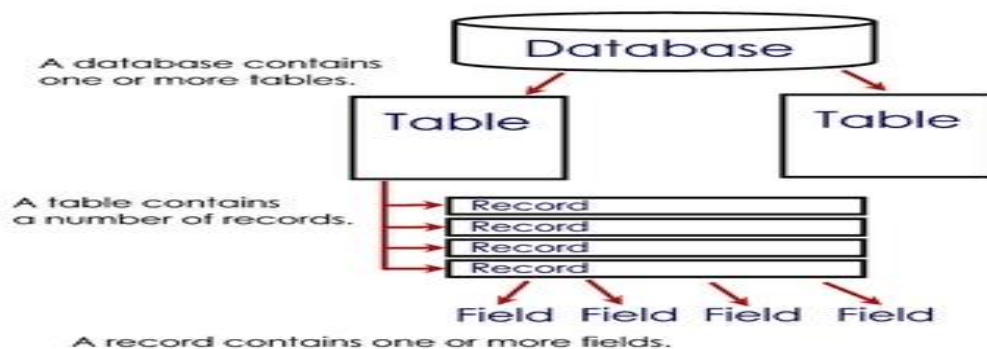
- Database
- Table
- Features of Database

Detailed content of the Lecture:

Characteristics of database:

- Provides security and removes redundancy
- Self-describing nature of a database system
- Insulation between programs and data abstraction
- Support of multiple views of the data
- Sharing of data and multiuser transaction processing
- Database Management Software allows entities and relations among them to form tables.
- It follows the ACID concept (Atomicity, Consistency, Isolation, and Durability).
- DBMS supports multi-user environment that allows users to access and manipulate data in parallel.

Organization of Database:



Characters:

- A byte of typically 8 bits.
- Each byte represent an characters.
- A Characters can be an Upper Case(A,B,C.....Z), Lower case(a,b,c,.....z), Numeric digit(0,1,2,.....9) are Special Symbol(.,!, [+], [-]/,.....).

Field:

- Character can be combined to form a field.
- Field is a single piece of information.
- It is defined as the smallest unit in a database.

Records:

- All the related files for an particular event are called a record.
- In another words records is one complete set of fields.

Table:

- Table is a collection of records.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=S_P2yyQuB8M

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (**Page No : 3 - 7**)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : I -Introduction.

Date of Lecture: 06.08.2020

Topic of Lecture: Data Models

Introduction :

- Data Model is a collection of concept that can be used to describe structure of the database.
- Structure of the database means data type relationship constraints that should hold the data.
- Data model the set of basic operation for specifying retrieve and update the database.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Tables
- Memory
- Design
- E-R Diagrams

Detailed content of the Lecture:

Categories of Data Models:

- Relational Model
- Network data Model
- Hierarchical Data Model
- Entity Relationship Model
- Object Oriented Data Model

Relational Model:

- This model represent data and relationship among data by a collection of tables known as relationship.
- A number of columns unique names.
- Relationship model relate records by the value that contain.

Network data Model:

- This model represent data by collection of records and relationship among data.
- This is represent by link which can be viewed as pointer.
- There are three components of network models,
 - Record Type
 - Data Elements
 - Link

Hierarchical Data Model:

- A DBMS to the hierarchical data model uses tree structure to relationship among records.

- A hierarchical database consist of a collection of records which are connected with each other through link.

Entity Relationship Model:

- The basic E-R diagram was conceived by Peter Chen in 1976.
- Its requirements in a top-down fashion.
- Diagram created using this process are called entity-relationship diagram or ER diagram.

Object Oriented Data Model:

- A object oriented data model is a logic organization real world object.
- A database language is concrete syntax for a data model.
- A database system implement the following basic object oriented concept,
 - Object and object identifies.
 - Attributes and methods.
 - Class.
 - Class hierarchical inheritance.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=4qFZ-5i4GS8>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 8 - 9)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



MCA

LECTURE HANDOUTS

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : I - Introduction

Date of Lecture:07.08.2020

Topic of Lecture: DBMS Architecture

Introduction :

- A DBMS is software designed to assist in maintaining and utilizing large collection of data.
- DBMS provides an environment that is both convenient and efficient to use.
- A DBMS is a combination of hardware and software that can be used to setup and monitor a database and manage the updating and retrieval of database.

Prerequisite knowledge for Complete understanding and learning of Topic:

- System
- Records
- Data models

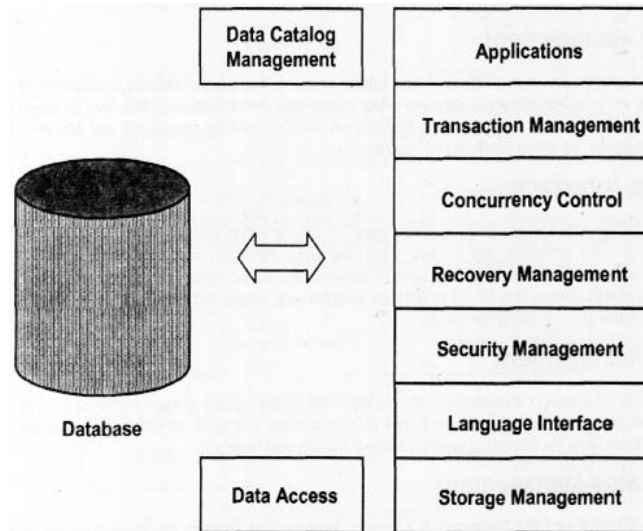
Detailed content of the Lecture:

Objective of DBMS:

- Shareability
- Availability
- Evolvability
- Integrity

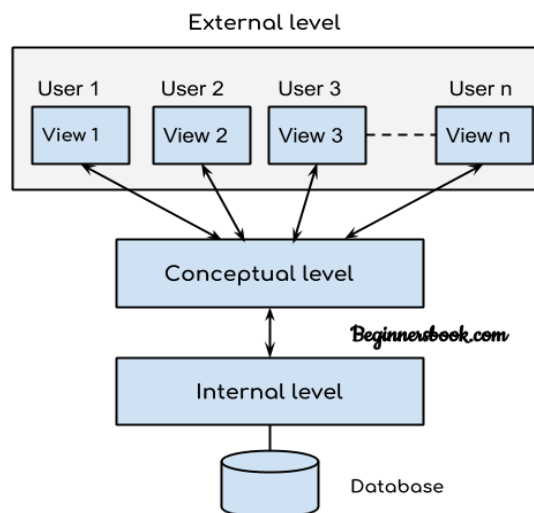
DBMS Interaction with Users:

- The database allows users to database without the knowledge.
- The communication between the DBMS software.



Data Architecture:

- A DBMS that provides three levels of data views called “three level architecture”.
- The objective of the three level architecture is to separate each user view of the database from the way the database is physically represented.
- External view.
- Conceptual view.
- Internal view.



External View:

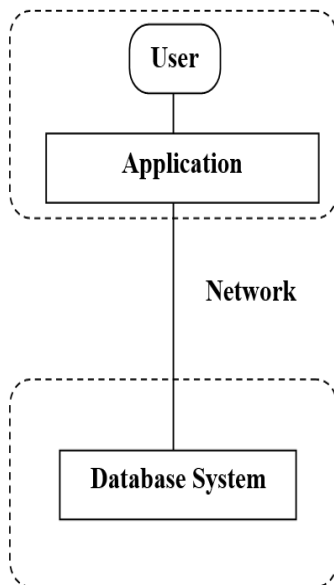
- The external level is at the highest level of database abstraction where only those portions of the database concurrence to a user are application program.
- Any number of user views may exist for a given global or conceptual view.
- Each external view is described by means of a schema called an external schema or sub-schema.
- The external schema consists of the logical records and the relationship in the external view.

Conceptual View:

- At the level of database abstraction all the database entities and the relationship.
- One conceptual view represented the entire database.
- The record and the relationship include in the conceptual view.
- The description of data at this level is in a format independent of its physical representation.

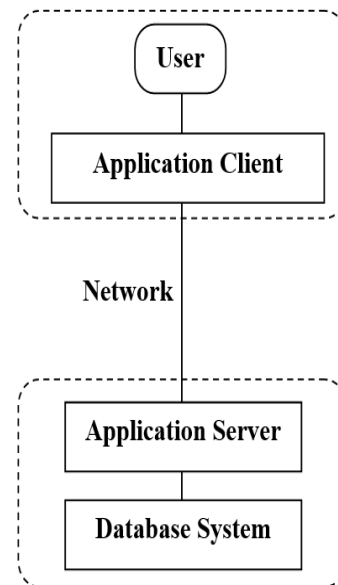
Internal View:

- The lowest level of abstraction closest to the physical storage method used.
- It the data will be stored and described the data structure and access methods to be used by database.
- The internet schema contains the storage record the method of representation the data field, expresses the internal view.



(a) Two-tier Architecture

client



Server

(a) Three-tier Architecture

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=dftMGbbULhE>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 28 - 31)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)

Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems /19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : I - Introduction

Date of Lecture:10.08.2020

Topic of Lecture: Data Independence

Introduction :

- Data Independency is the capacity to change the schema at one level of database system without changing the schema at the next higher level.
- Immunity to lower level implementation details.
- Possibility of adding new application without having to restructure the database

Prerequisite knowledge for Complete understanding and learning of Topic:

- Relationship
- Attributes
- Data Structures

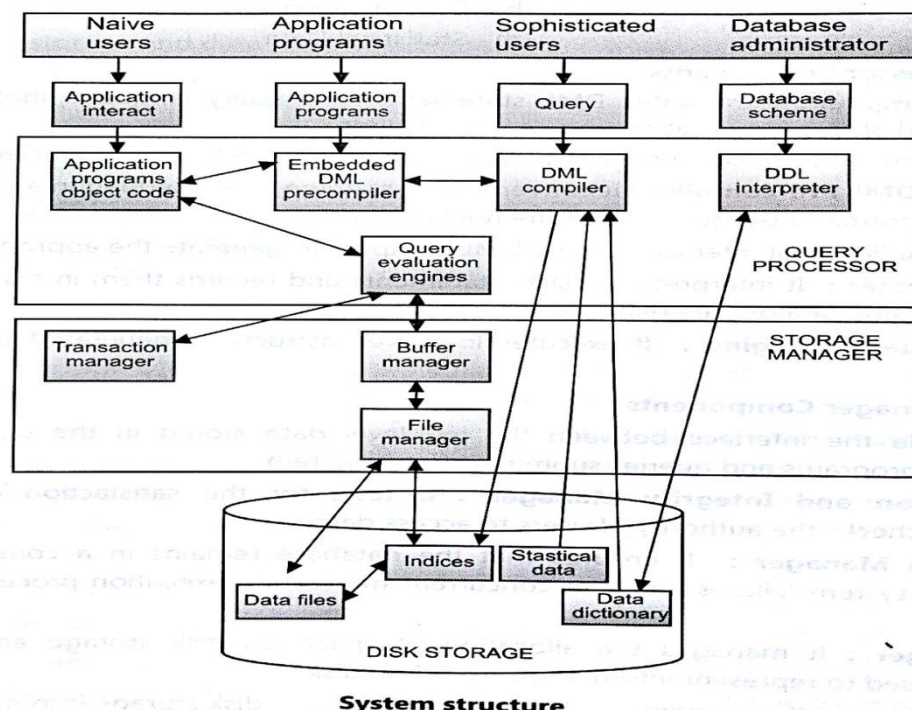
Detailed content of the Lecture:

Categories of Data Independence:

- Physical Data Independence
- Logical Data Independence

Advantages Data Independence:

- Redundancy and inconsistency can be reduced.
- Better service to user.
- Cost of developing and maintaining system is lower.
- Security can be improved.
- Enterprise requirement can be identified.



DML Pre-Compiler:

- The DML(Data Manipulation Language) statement embedded in an application program to normal produce calls in the host languages.
- The pre-compiler must interact with the query processor in order to generate the approximated quote code.

DDL Interpreter:

- The DDL(Data Definition Languages) interpreter converts the definition statement into a set of table.

File Manager:

- File manager manage the allocation of space on disk storage and the data structure used to represent information stored on disk.

Database Manager:

- The database manager is a program module which provides the interface between low level data stored.
- The database and the application programs and query to be submitted to the system.

Query Processor:

- The query is used to interpret the online user query and convert into an efficient service of operation.
- This the sent of the data manager for exception.

Database Administrator:

- Database Administration is a high level function that is responsible for overall management of data resources in an organization.

Advantages Of DBMS:

- Redundancy can be reduced.
- Inconsistency can be avoided.
- Data can be shared.
- Standard can be enforced.
- Security can be applied.
- Integrity can be maintained
-

Disadvantages of DBMS:

- High cost.
- Problem associated with centralization.
- Complexity of backup and recovery.
- Confidentiality privacy and security.
- Data quality.

Database Application:

- Airlines.
- Education.
- Banking.
- Telecommunications.
- Credit card Transaction.
- E-Commerce.
- Finance.
- Sales.
- Human Resources

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=dftMGbbULhE>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (**Page No : 24 - 28**)



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : I - Introduction

Date of Lecture: 11.08.2020

Topic of Lecture: Data Modeling using Entity

Introduction :

- The basic E-R diagram was conceived by Peter Chen in 1976.
- Its requirements in a top-down fashion.
- Diagram created using this process are called entity-relationship diagram or ER diagram.
- It is simple and easy to understand with a minimum of training.
- The model can be used by the database designer to communicate the designer to the end user.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Records
- Dependence
- Data design
- Architecture

Detailed content of the Lecture:

Entity:

- Entity is the object that exist and is distinguishable from other object.
- **EX:**
- A particular person.
- A particular department.
- A particular place.

Entity Set:

- Entity set is a collection or set all entities of a particular entity type at any point of time is called entity set or extension of a entity type.
- **EX:**
- Bucket of apple, etc.....,

Types of Entity set:

- Strong Entity Set
- Weak Entity Set

Strong Entity Set:

- A strong entity set is one that has a complete identifies values may be used to identifying instance uniquely.
- A weak entities is dependent on strong entity.

Weak Entity Set:

- Entity belong to a weak entity type are identified by being related to specify entities from another entity type in communication with some of their attributes values.

ATTRIBUTES:

- A Particular entity will have a value for each of its attributes.
- The attributes values that describe each entity become a major part of the data stored in the database.

EX:

- An employee entity may be described by the employees, name, age, address, salary and job

Types of Attributes:

- Composite Attributes.
- Simple Attributes.
- Single and Multi Valued.
- Stored and Derived Attributes.

KEYS:

- A key is single attribute are combination of two or more attributes of an entity set that is used to identify one or more instances of set.

Types of Keys:

- Primary Key.
- Super Key.
- Candidate Key.
- Foreign Key.
- Alternate Key.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=x5dOJK4V_Rk

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 204 - 210)



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)

Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



MCA

LECTURE HANDOUTS

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : I - Introduction

Date of Lecture: 12.08.2020

Topic of Lecture : Relationship Model

Introduction :

- There are two entities, both of them are of the person type.
- There is a relationship called married between the two person.
- In this relationship each of these two person entity has a role.
- One person play the role of husband and another person plays the role of wife.

Prerequisite knowledge for Complete understanding and learning of Topic:


- Tables
- Memory
- Design
- E-R Diagrams


Detailed content of the Lecture:


Mathematical Relationship on Entity Set:

- A relationship type R among n entity types R_1, R_2, \dots, R_n define a set of association among entities from this types.
- Mathematically R is a set of relationship instance r_i associate an n entities (e_1, e_2, \dots, e_n) and each entity E_i in r_i is a member of entity type.

Types of relationship:

□ one to one: 

□ one to many: 

□ many to many: 

One-to-One:

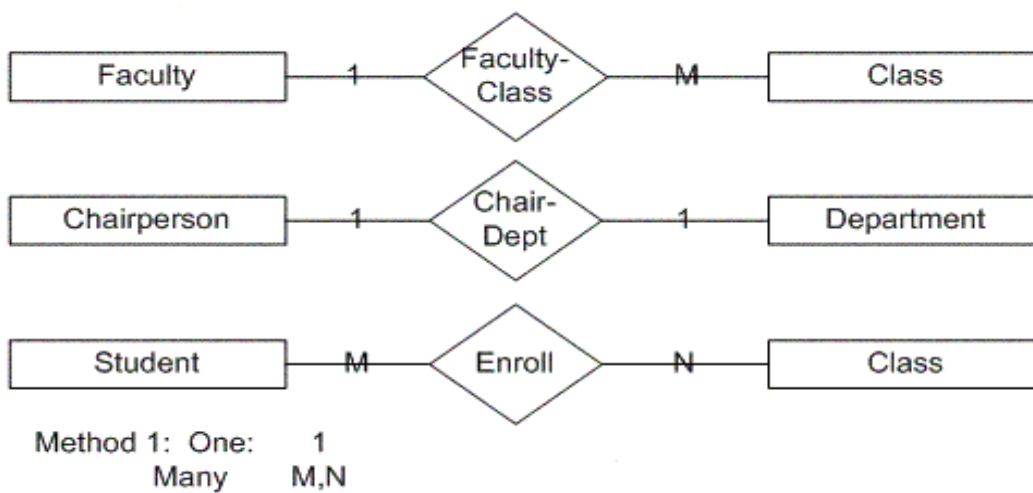
- The one-to-one relationship also denoted as 1:1 is simple.
- An entity there is exactly one occurrence of another entity.

One-to-Many:

- In a one-to-many relationship also denoted as 1:M for given occurrence of an entity there can be one or more occurrence of another entity.

Many-to-Many:

- In a many-to-many relationship also denoted as M:N there can be one or more values on both relationship of values



Degree of Relationship:

- Unary Relationship.
- Binary Relationship.
- Ternary Relationship.
- Quaternary Relationship.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=p7GsjrXoW28>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 37 - 42)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



MCA

LECTURE HANDOUTS

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : I - Introduction

Date of Lecture: 13.08.2020

Topic of Lecture: Structural Constraints

Introduction :

- Structural constraints are information about how two or more entities are related to one another.
- Structural Constraints are also called Structural properties of a database management system (DBMS).
- The name constraints refer to the fact that such limitations must be imposed on the data, for the DBMS system to be consistent with the requirements.

Prerequisite knowledge for Complete understanding and learning of Topic:

- SQL
- Tables
- Triggers
- Records

Detailed content of the Lecture:

Types of Structural Constraints:

- Cardinality Constraints/Ratio
- Participation Constraints

Cardinality Constraints/Ratio:

- The cardinality specify the maximum number of relationship instant that the entity can participate. The term used in database relation to denote the occurrence on data on either side of the relationship.
- The possible cardinality ratio binary relationship types are,
 - 1 : 1
 - 1 : M
 - M : N

Types of Cardinality:

- Link Cardinality.
- Sub-type Cardinality.
- Physical segment Cardinality.
- Possession Cardinality.
- Child Cardinality.
- Characteristic Cardinality.
- Paradox Cardinality.
- Association Cardinality.

Participation Constraints:

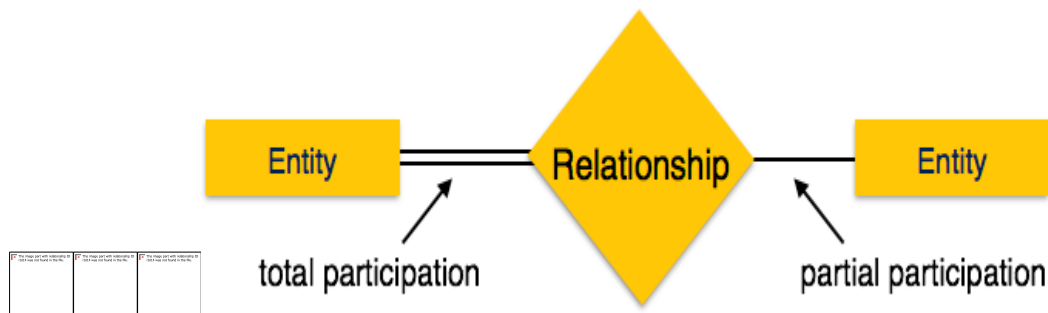
- A Participation constraints specifies the existence of an entity depends on its related to another
- Total Participation
- Partial Participation

Total Participation:

- Total Participation- It specifies that each entity in the entity set must compulsorily participate in at least one relationship instance in that relationship set

Partial Participation:

- It's one or more attributes that uniquely identify a weak entity for a given owner entity.
- It's called partial because it can't be a primary key on it's own, it needs another column, which is the foreign key of the owner entity.



Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=RrjNHZ-EEOI>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 210- 212)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : I - Introduction

Date of Lecture: 14.08.2020

Topic of Lecture: Enhanced E-R Modeling.

Introduction :

- Enhanced entity-relationship models, also known as extended entity-relationship models, are advanced database diagrams very similar to regular ER diagrams.
- Enhanced ERDs are high-level models that represent the requirements and complexities of complex databases

Prerequisite knowledge for Complete understanding and learning of Topic:

- Tools
- Database
- E-R Diagrams
- Data Design

Detailed content of the Lecture:

Features of EER Model

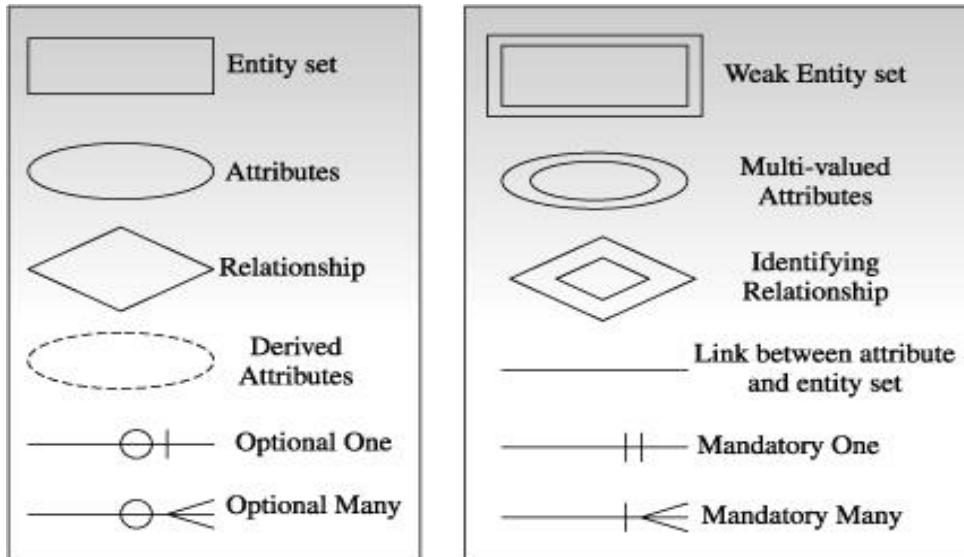
- EER creates a design more accurate to database schemas.
- It reflects the data properties and constraints more precisely.
- It includes all modeling concepts of the ER model.
- Diagrammatic technique helps for displaying the EER schema.
- It includes the concept of specialization and generalization.
- It is used to represent a collection of objects that is union of objects of different of different entity type.

Super Class:

- Super class is an entity type that has a relationship with one or more subtypes.
- An entity cannot exist in database merely by being member of any super class.
- **For example:** Shape super class is having sub groups as Square, Circle, Triangle.

Sub Class:

- Sub class is a group of entities with unique attributes.
- Sub class inherits properties and attributes from its super class.
- **For example:** Square, Circle, Triangle are the sub class of Shape super class.



Specialization and Generalization:

Generalization:

- Generalization is the process of generalizing the entities which contain the properties of all the generalized entities.
- It is a bottom approach, in which two lower level entities combine to form a higher level entity.
- Generalization is the reverse process of Specialization.
- It defines a general entity type from a set of specialized entity type.
- It minimizes the difference between the entities by identifying the common features.
- **For example:**

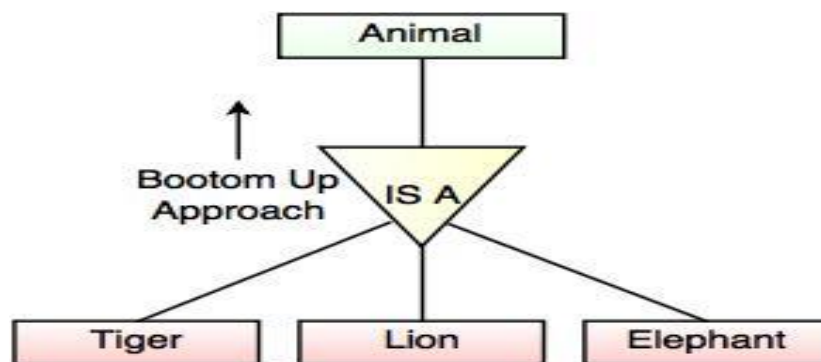


Fig. Generalization

- In the above example, Tiger, Lion, Elephant can all be generalized as Animals.

Specialization:

- Specialization is a process that defines a group entities which is divided into sub groups based on their characteristic.
- It is a top down approach, in which one higher entity can be broken down into two lower level entity.
- It maximizes the difference between the members of an entity by identifying the unique characteristic or attributes of each member.

- It defines one or more sub class for the super class and also forms the superclass/subclass relationship.

For example : In the above example, Employee can be specialized as Developer or Tester, based on what role they play in an Organization.

Aggregation:

- Aggregation is a process that represent a relationship between a whole object and its component parts.
- It abstracts a relationship between objects and viewing the relationship as an object.
- It is a process when two entity is treated as a single entity.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=MTG1zl8PkXk>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (**Page No : 227 - 236**)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



MCA

LECTURE HANDOUTS

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : II - Relational Model And Query Evaluation

Date of Lecture: 17.08.2020

Topic of Lecture: Relational Model Concepts

Introduction :

- The relational data model was 1st introduced in 1970 by EF codd.
- The relational model is the most widely used method of organizing data.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Data Structures
- Data Model
- Data Algorithm

Detailed content of the Lecture:

Relational Model Concepts:

- The relational data model was 1st introduced in 1970 by EF codd. The relational model is the most widely used method of organizing data.
- The relational model produce a simple and flexible way.
- It is even visible in day-to-day life.
- The relational data model represent data in the form of tables.
- The data Structure.
- Data manipulation.
- Data Integrity.

Data Structure:

- Data are organized in the form of tables with rows and columns.

Data Manipulation:

- Powerful operations are used to manipulate the data stored in the relations.

Data Integrity:

- Facilities are include to specify business rules that maintain the integrity of data manipulated.

Relations:

- A relation is two-dimensional table of data.
- Each relation consist of a set of named columns and an arbitrary number of unnamed rows.
- An attributes consist with its named column of a relation.

- Each row of a relation corresponded to a records that contain data values for a single entity.

Properties of Relations:

- **No Duplicate Tuples :** A relation may contain two or more tuples which have the same value for all the attributes.
- **Tuples are unordered:** The ordered of the rows in a relation is material.
- **Attributes are Unordered :** The order of the column in the relation is unordered.
- **Attributes values are Atomic:** Each tuple contained exactly one value for each attributes.

Advantages of Relation Model:

- Easy to use
- Flexibility
- Precision

Disadvantages of Relational Model:

- Hardware Overheads.
- Ease of design can result in bad design.
- Information Island Phenomenon.

CODD Rules:

- **Foundation Rule :** A relational database management system must manage its stored data using only its relational capabilities.
- **Information Rule :** All information in the database should be represented in one and only one way as values in a table.
- **Guaranteed Access Rule :** Each and every data is to be logically accessible by restoring to a combination of table name, primary key value and column name.
- **Systematic treatment of Null Values:** Null values are supported in the fully relational DBMS for representing missing information in a systematic way independent of data type.
- **Dynamic On -line catalog based on the Relational model:** The database description is represented at the logical level in the same way as ordinary data.
- **Comprehensive data sub language Rule:** A relational system may support several languages and various modes of terminal use.
- **View Updating Rule :** All views that are theoretically updateable are also updateable by the system. High-level Insert, Update, and Delete: The capability of handling a base relation or a derived relation as a single operand apply not only to the retrieve of data but also do the insert, update, deletion of data .
- **Physical data independence:** Application programs and terminal activities remains logically unimpaired any change are made in either storage represent are access method.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=P8n_rwPzdBc

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hill, Sixth Edition, 2010 (Page No : 37 - 40)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



MCA

LECTURE HANDOUTS

I / 1

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs.R.Pavithra

Unit : II - Relational Model And Query Evaluation

Date of Lecture: 24.08.2020

Topic of Lecture: Relational Algebra

Introduction :

- Relational Algebra is the mathematical basic for relational database developed by E.F.Codd.
- Relational Algebra is a notation for representing various operations which can be performed on relational database.

Prerequisite knowledge for Complete understanding and learning of Topic:

- SQL Query
- Relationship
- Data Manipulation

Detailed content of the Lecture:

Relational Algebra:

- The relational algebra is a Procedure Query Language
- It consist of a set of operations that take one or two relations as input and produce a new relation as the result.

Fundamental relational algebra operation:

- The fundamental operations take relation as operands and produce relations as results which are the mathematical property.
- Unary Operation
- Binary Operation

Unary Operation:

- The unary operations which operate on one operands.
- EX: Select, project, rename, operation.

Binary Operation:

- There are operations which operate on two operands known as binary operation.
- EX: Union, Set difference, etc.....,

Types of Relational Algebra Operation:

- Select Operation (σ)
- Project Operations(Π)
- Rename Operations(ρ)
- Union Operations(U)
- Set difference Operations (-)
- Cartesian-Product Operations(X)

Select Operations:

- The select operation select tuples that satisfy a given predicate.
- Greek letter (σ) sigma is used to indicate the select operation.
- It is a horizontal subset of a given relation.
- The predicate is expressed as a Boolean combination of terms, each terms begin a simple comparison that can be established as true or false for a given tuple.

Project Operation(Π):

- Projection operator has a vertical subset of a given relation that is a subset obtained in a specified left to right order and them eliminating duplicate tuples of the attributes selected.
- Project operations is denoted by the Greek letter Pi(Π).
- The project operations is a unary operation that returns its argument relations.
- A relation is set any duplicate row are eliminated.

Rename Operations(ρ):

- The rename operation of a relation instance that are defined by the relational algebra expression must be explicitly named.
- That the name large algebraic expressions could be easily broke down into small pieces by giving name to the result of sub expression.
- Thus ρ is introduced as a renaming operation.
- In relational algebra a rename is a Unary operation return as $\rho_{a/b}(R)$.
- Where, a and b attributes names $R \longrightarrow$ relation

Union Operations(U):

- The union operation on relation A union relation B designed as $A \cup B$, join are include all tuples that are in A (or) B eliminating duplicate tuples.
- **EX:** Π customer-name(Borrower)
- Finding the name of all customers with an account in the bank.
- Π customer –name (Depositor)
- $(\Pi$ customer – name (BPRROWER)) U $(\Pi$ customer-name (DEPOSITOR))

Set difference Operations (-):

- It finding tuples that are in one relation but are not in another . The set difference operation denoted by “-” .
- The expression A-B result in a relation containing those tuples in A but not in B.
- Π customer – name (DEPOSITER)- Π customer –Name (BORROWER).

Cartesian-Product Operations(X):

- The Cartesian product operation is similar to set.
- It is a relation consisting of all possible pairing of tuples.
- Assume one has relational.

$R(A_1, A_2, A_3, \dots, A_N)$ and Then,
 $(B_1, B_2, B_3, \dots, B_M)$

$RXS = \{(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m) / (A_1, A_2, \dots, A_n) \in R \text{ AND } (B_1, B_2, \dots, B_m) \in S\}$

- RXS is not the same as SXR are because the order of attributes differ.

Additional Relational Algebra Operation:

- This operations do not add any power to the algebra but they simplify common queries.
- Set-intersection(binary)
- Join(Binary)
- Division(Binary)
- Assignment(Unary)

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=4YilEjkNPrQ>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 46 - 59)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE
(An Autonomous Institution)



(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to
Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L 12

MCA

I / I

Course Name with Code : Relational Database Management System / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : II - Relational Model And Query Evaluation

Date of Lecture: 25.08.2020

Topic of Lecture: SQL , Basic Queries , Complex SQL Queries

Introduction :

- SQL is short for Structured Query Language(SQL) is an international standard language for creating or manipulating relational database.
- To communicate with the RDBMS itself user uses the SQL.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Data Definition Language
- Data Manipulation Language
- Transaction Control Language

Detailed content of the Lecture:

SQL:

- SQL can create schema, delete them and change them.
- It can also put data into schema and remove data.
- It is a data handling language but it is not a programming language.

Elements of SQL Language:

- **Clauses :** There are some clauses optional, constituent compounds of statement and queries.
- **Expression :** This can produce either scalar value or tables consisting of columns and rows of data.
- **Predicates:** It specify condition that can be evaluated to SQL 3 valued logic(3VL) . Boolean truth values and which are used to limit the effects of statements or queries to state change program flow.
- **Queries:** Retrieve data based to specific criteria.
- **Statement:** The predicates effects on schema and data . The control transactions, program connections, section or diagnostics.
- A SQL statement also included the semicolon(;).
-

Advantages of SQL:

- Portable
- Easy to learn and understand.
- Used with any DBMS system with any vender.
- Interactive language.

- Supports object based programming.

Disadvantages of SQL :

- Vendor variations.
- Join Inconsistencies.
- Misconstructions in where.
- Complex grammar.

SQL data types:

- **Character(n):** The data type represent a fixed length string of exactly n characters.
- N is greater than zero and should be an integer.
- **Character varying(n):** This data type represent a varying length string values maximum length is n characters.
- N is a +ve integer and “VARCHAR” is a abbreviation for character varying.
- **Bit(n) and Bit varying(n):** Bit(n) represent a fixed length string of exactly n bit.
- Bit varying represent a varying length string whose maximum length can be n bits.
- Both case n must be +ve integer.
- **Numeric(p,q):** This data type represent a decimal number p digit and sign with assumed decimal point Q digit from the right.
- Both P and Q are integer.
- P should be greater than 0.
- Q can be equal to 0.
- Q should can be less than or equal to P
- **Decimal(p,q):** Decimal(P,Q) represent a decimal number.
- M digit and sign with assumed decimal point Q digit from the right.
- $0 \leq q < p < m, p > 0$
- ‘p’, ‘q’ and ‘m’ should be integers.
- DEC is an abbreviation of decimal.
- **Integer:** Integer represent as a signed integer decimal or binary.
- Int is a abbreviation of integer.
- **Small int:** This data type represent a signed integer decimal or binary.
- The integer is decimal or binary is implementation is defined.
- The actual int and small int is implementation defined.
- **Float(p):** Float(p) represent a floating point number.
- Float is abbreviation for float(p).

SQL Commands:

- SQL commands or lines of SQL codes that the SQL application to perform simple task against as with the data inside of the database.

Types of SQL Commands:

- Data Definition Language(DDL).
- Data Manipulation Language(DML).
- Data Query Language(DQL).
- Data Control Language(DCL).
- Data Administration Statement(DAS).
- Transaction Control Statement(TCS).

Data Definition Language(DDL):

- Data definition language is used to create, alter and delete database object.
- There are 3 manipulation commands.
- Create table.
- Alter table.
- Drop table.

Data Manipulation Language(DML):

- DML commands user insert data into the database.

- Modify and delete the data in the database.
- Insert.
- Update.
- Delete.

Data Query Language(DQL):

- This is one of the most commonly used SQL statement.
- This SQL enable the user to Query one or more table to get the information we want to the SQL
- Select

Data Control Language(DCL):

- The DCL is prevent unauthorized access to data.
- The database administrator power to give the task the privilege to a specific user.
- Grant.
- Revoke.

Data Administrator Statement(DAS):

- The commands the user to perform adults to analysis on operation within the database.
- Start Audit.
- Stop Audit.

Transaction Control Statements(TCS):

- TCS manage all the changes made by the DML statement.
- Commit.
- Roll back.
- Save point.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=FNYdBLwZ6cE>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 80 - 85)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : II - Relational Model And Query Evaluation

Date of Lecture:28.08.2020

Topic of Lecture: Views, Constraints

Introduction :

- A database view is a subset of a database and is based on a query that runs on one or more database tables.
- Constraints are the rules enforced on the data columns of a table. These are used to limit the type of data that can go into a table.

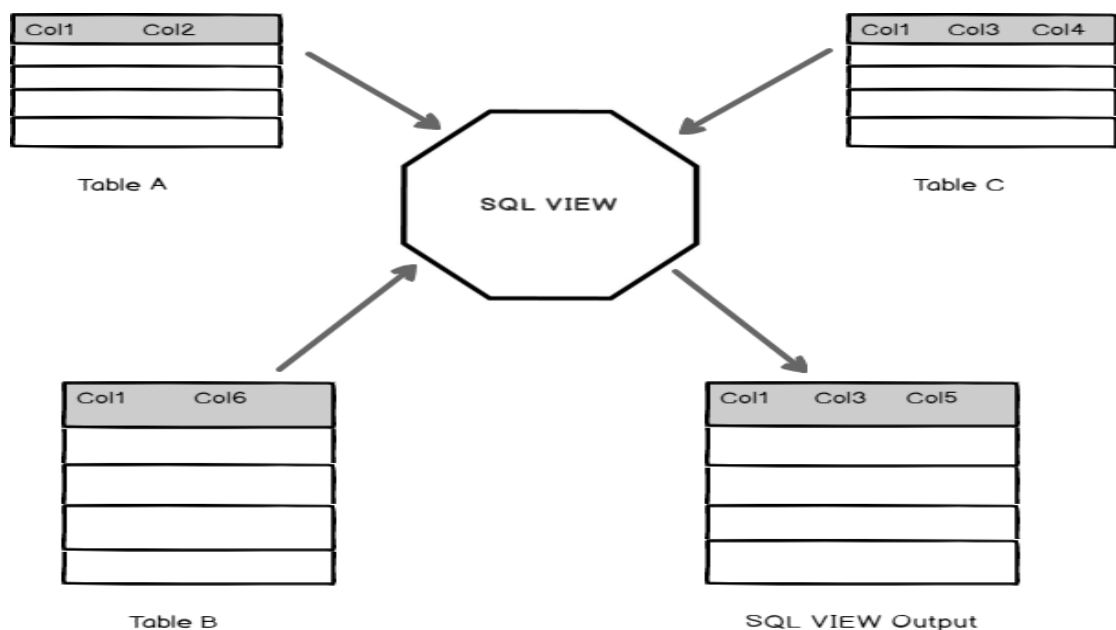
Prerequisite knowledge for Complete understanding and learning of Topic:

- Table
- Query
- Database

Detailed content of the Lecture:

Views:

- A view in SQL Server is like a virtual table that contains data from one or multiple tables. It does not hold any data and does not exist physically in the database.



Creating View :

- Views in SQL are kind of virtual tables.
- A view also has rows and columns as they are in a real table in the database.
- We can create a view by selecting fields from one or more tables present in the database.
- A View can either have all the rows of a table or specific rows based on certain condition.

Syntax:

```
CREATE VIEW view_name AS  
SELECT columns  
FROM tables  
[WHERE conditions];
```

Updating a View:

- The SELECT clause may not contain the keyword DISTINCT.
- The SELECT clause may not contain summary functions.
- The SELECT clause may not contain set functions
- The SELECT clause may not contain set Operations.
- The SELECT clause may not contain an ORDER BY clause.
- The FROM clause may not contain multiple tables.
- The WHERE clause may not contain subqueries.

Advantages of Views:

- Restrict Data Access and/or Simplify Data Access.
- Simplify Data Manipulation.
- Import and Export Data.
- Merge Data

Constraints:

- SQL constraints are **used to specify rules for the data in a table.**
- Constraints are used to limit the type of data that can go into a table.
- This ensures the accuracy and reliability of the data in the table.

Types of Constraints:

- Unique Constraint.
- Primary Constraint.
- Default Constraint.
- Check Constraint.
- Foreign Constraint.

Unique Constraint:

- The UNIQUE constraint ensures that all values in a column are different.

Syntax:

```
CREATE TABLE Employee  
(Ecode integer, NOT NULL UNIQUE,  
Ename char(20), NOT NULL,  
Grade char(2))
```

Gross decimal);

Primary Constraint:

- A primary key is a field in a table which uniquely identifies each row/record in a database table.
- Primary keys must contain unique values.
- A primary key column cannot have NULL values.

Syntax:

```
CREATE TABLE Employee
(Ecode integer, NOT NULL PRIMARY KEY,
Ename char(20), NOT NULL,
Grade char(2)
Gross decimal);
```

Default Constraint:

- The DEFAULT Constraint is used to fill a column with a default and fixed value.

Syntax:

```
CREATE TABLE Employee
(Ecode integer, NOT NULL PRIMARY KEY,
Ename char(20), NOT NULL,
Grade char(2) DEFAULT='E1'
Gross decimal);
```

Check Constraint:

- The CHECK constraint is used to limit the value range that can be placed in a column.

Syntax:

```
CREATE TABLE Employee
(Ecode integer, NOT NULL PRIMARY KEY,
Ename char(20), NOT NULL,
Grade char(2) DEFAULT='E1'
Gross decimal CHECK (gross>2000);
```

Foreign Constraint:

- Foreign keys are the columns of a table that points to the primary key of another table.
- They act as a cross-reference between tables.

Syntax:

```
Columnname datatype (size) REFERENCE tablename
[(Columnname)] [ON DELETE CASCADE]
```

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=mBY-Dirn2MU>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 210 - 214)



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : II - Relational Model And Query Evaluation

Date of Lecture:29.08.2020

Topic of Lecture: Relational Calculus

Introduction :

- Relational algebra, we now consider an applied predicate calculus which may also be used in the formulation of queries on any data base consisting of a finite collection of relations in simple normal form.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Tuple
- Queries
- Relational Calculus

Detailed content of the Lecture:

Relational Calculus:

- Two types of Relational Calculus,
- Tuple Relational Calculus (TRC)
- Domain Relational Calculus (DRC)

Tuple Relational Calculus (TRC):

- Tuple Relational Calculus is a non-procedural query language unlike relational algebra.
- Tuple Calculus provides only the description of the query but it does not provide the methods to solve it
- In Tuple Calculus, a query is expressed as

$$\{t | P(t)\}$$

where t = resulting tuples,

P(t) = known as Predicate and these are the conditions that are used to fetch t

Thus, it generates set of all tuples t, such that Predicate P(t) is true for t.

P(t) may have various conditions logically combined with OR (\vee), AND (\wedge), NOT (\neg).

It also uses quantifiers:

$\exists t \in r (Q(t)) =$ "there exists" a tuple in t in relation r such that predicate Q(t) is true.

$\forall t \in r (Q(t)) =$ Q(t) is true "for all" tuples in relation r.

Domain Relational Calculus (DRC):

- Domain Relational Calculus is a non-procedural query language equivalent in power to Tuple Relational Calculus.
- Domain Relational Calculus provides only the description of the query but it does not provide the methods .
- In Domain Relational Calculus, a query is expressed as,

$$\{ \langle x_1, x_2, x_3, \dots, x_n \rangle \mid P(x_1, x_2, x_3, \dots, x_n) \}$$

where, $\langle x_1, x_2, x_3, \dots, x_n \rangle$ represents resulting domains variables and $P(x_1, x_2, x_3, \dots, x_n)$ represents the condition or formula equivalent to the Predicate calculus.

Predicate Calculus Formula:

1. Set of all comparison operators
2. Set of connectives like and, or, not
3. Set of quantifiers

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=4FYN_mErztg

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 163- 171)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : II - Relational Model And Query Evaluation

Date of Lecture: 30.08.2020

Topic of Lecture: Dynamic and embedded SQL

Introduction :

- Embedded SQL are SQL statements in an application that do not change at runtime and, therefore, can be hard-coded into the application.
- Dynamic SQL is SQL statements that are constructed at runtime.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Tables
- Records
- Files

Detailed content of the Lecture:

Dynamic and embedded SQL:

- Dynamic SQL is a programming technique that enables you to build SQL statements dynamically at runtime.
- For example, dynamic SQL lets you create a procedure that operates on a table whose name is not known until runtime.

Embedding SQL statements in a host language:

- Structured Query Language (SQL) is a standardized language that you can use to manipulate database objects and the data that they contain.
- Despite differences between host languages, embedded SQL applications are made up of three main elements that are required to setup and issue an SQL statement.

Supported development software for embedded SQL applications:

- Before you begin writing embedded SQL applications, you must determine if your development software is supported.
- The operating system that you are developing for determines which compilers, interpreters, and development software you must use.

Setting up the embedded SQL development environment:

- Before you can start building embedded SQL applications, install the supported compiler for the host language you will be using to develop your applications and set up the embedded SQL environment.

Designing embedded SQL applications:

- When designing embedded SQL applications you must use static or dynamic executed SQL statements.

Programming embedded SQL applications:

- Programming embedded SQL applications involves the same steps required to assemble an application in your host programming language.

Building embedded SQL applications:

- After you have created the source code for your embedded SQL application, you must follow additional steps to build the application.
- You should consider building 64-bit executable files when developing new embedded SQL database applications. Along with compiling and linking your program, you must precompile and bind it.

Deploying and running embedded SQL applications:

- Embedded SQL applications are portable and can be placed in remote database components.
- You can compile the application in one location and run the package on a different component.

Compatibility features for migration:

- The Db2 database manager provides features that facilitate the migration of embedded SQL C applications from other database systems.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=C0UqMaCxyAk>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 134- 137)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : II - Relational Model And Query Evaluation

Date of Lecture: 01.09.2020

Topic of Lecture: Database Design

Introduction :

- Database design is a collection of steps that help create, implement, and maintain a business's data management systems.
- The primary purpose of designing a database is to produce physical and logical models of designs for the proposed database system.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Physical Model
- Logical Model
- Relationship

Detailed content of the Lecture:

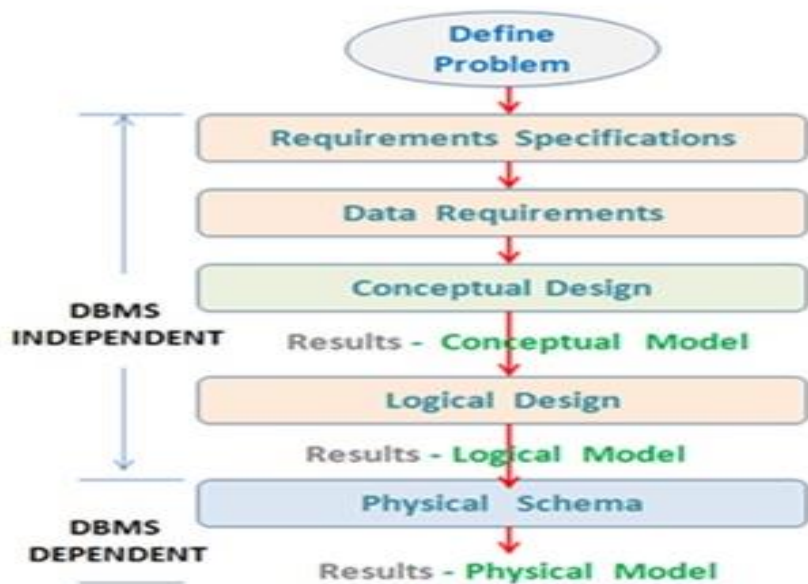
Database Design:

- Database design is the organization of data according to a database model.
- The designer determines what data must be stored and how the data elements interrelate.
- With this information, they can begin to fit the data to the database model.[1] Database management system manages the data accordingly.
- Database design involves classifying data and identifying interrelationships.
- This theoretical representation of the data is called an *ontology*. The ontology is the theory behind the database's design.

Database design Approaches:

- Top-Down Database Design Approach.
- Bottom-up Database Approach.
- Inside-Out Database Design Approach.
- Mixed Strategy Database Design Approach.

Database Design Process:



--	--	--

Anomalies in a Database:

- Redundant Storage.
- Update Anomaly.
- Insertion Anomaly.
- Deletion Anomaly.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=ztHopE5Wnpc>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 201- 203)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : II - Relational Model And Query Evaluation

Date of Lecture:03.09.2020

Topic of Lecture: Functional Dependencies

Introduction :

- A functional dependency is a constraint that specifies the relationship between two sets of attributes where one set can accurately determine the value of other sets.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Attributes.
- Data sets
- Relationship
- Constraints

Detailed content of the Lecture:

Functional Dependencies :

- A functional dependency is a constraint that specifies the relationship between two sets of attributes where one set can accurately determine the value of other sets.
- It is denoted as $X \rightarrow Y$, where X is a set of attributes that is capable of determining the value of Y.
- The attribute set on the left side of the arrow, X is called Determinant, while on the right side, Y is called the Dependent.
- Functional dependencies are used to mathematically express relations among database entities and are very important to understand advanced concepts in Relational Database System and understanding problems in competitive exams like Gate.

Armstrong's axioms/properties of functional dependencies:

- **Reflexivity:** If Y is a subset of X, then $X \rightarrow Y$ holds by reflexivity rule
For example: {roll_no, name} \rightarrow name is valid.
- **Augmentation:** If $X \rightarrow Y$ is a valid dependency, then $XZ \rightarrow YZ$ is also valid by the augmentation rule.
For example: If {roll_no, name} \rightarrow dept_building is valid, hence {roll_no, name, dept_name} \rightarrow {dept_building, dept_name} is also valid. \rightarrow

- **Transitivity:** If $X \rightarrow Y$ and $Y \rightarrow Z$ are both valid dependencies, then $X \rightarrow Z$ is also valid by the Transitivity rule.
For example: $\text{roll_no} \rightarrow \text{dept_name}$ & $\text{dept_name} \rightarrow \text{dept_building}$, then $\text{roll_no} \rightarrow \text{dept_building}$ is also valid.

Types of Functional dependencies in DBMS:

- Trivial functional dependency
- Non-Trivial functional dependency
- Multivalued functional dependency
- Transitive functional dependency

Trivial Functional Dependency:

- In Trivial Functional Dependency, a dependent is always a subset of the determinant.
- i.e. If $X \rightarrow Y$ and Y is the subset of X , then it is called trivial functional dependency.

Non-trivial Functional Dependency:

- In Non-trivial functional dependency, the dependent is strictly not a subset of the determinant.
- i.e. If $X \rightarrow Y$ and Y is not a subset of X , then it is called Non-trivial functional dependency.

Multivalued Functional Dependency:

- In Multivalued functional dependency, entities of the dependent set are not dependent on each other.
- i.e. If $a \rightarrow \{b, c\}$ and there exists no functional dependency between b and c , then it is called a multivalued functional dependency.

Transitive Functional Dependency

- In transitive functional dependency, dependent is indirectly dependent on determinant.
- i.e. If $a \rightarrow b$ & $b \rightarrow c$, then according to axiom of transitivity, $a \rightarrow c$. This is a transitive functional dependency

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=dR-jJimWWHA>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 293- 296)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : II - Relational Model And Query Evaluation

Date of Lecture: 04.09.2020

Topic of Lecture: Normalization

Introduction :

- Normalization is the process of organizing data in a database.
- This includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Forms
- Records
- Database
- Redundancy

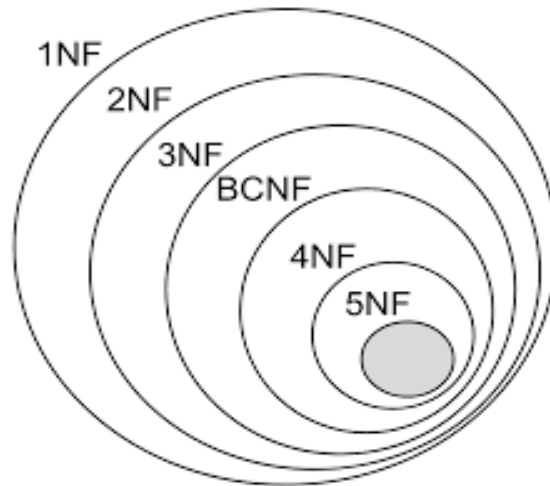
Detailed content of the Lecture:

Normalization:

- Normalization entails organizing the columns (attributes) and tables (relations) of a database to ensure that their dependencies are properly enforced by database integrity constraints.
- It is accomplished by applying some formal rules either by a process of synthesis.

Normal Forms:

- Normalization is the process of minimizing redundancy from a relation or set of relations.
- Redundancy in relation may cause insertion, deletion and updation anomalies
- So, it helps to minimize the redundancy in relations. Normal forms are used to eliminate or reduce redundancy in database tables
- 1NF: First normal form.
- 2NF: Second normal form.
- 3NF: Third normal form.
- BCNF: Boyce–Codd normal form.
- 4NF: Fourth normal form.
- 5NF Fieth Normal Form.



1NF: First normal form:

- The first normal form states that: Every column in the table must be unique. Separate tables must be created for each set of related data.
- Each table must be identified with a unique column or concatenated columns called the primary key

A table is in 1 NF iff:

- There are only Single Valued Attributes.
- Attribute Domain does not change.
- There is a unique name for every Attribute/Column.
- The order in which data is stored does not matter.

2NF: Second normal form:

- Second Normal Form (2NF) is based on the concept of full functional dependency.
- Second Normal Form applies to relations with composite keys, that is, relations with a primary key composed of two or more attributes.
- A relation with a single-attribute primary key is automatically in at least 2NF. A relation that is not in 2NF may suffer from the update anomalies.

3NF: Third normal form:

- Third normal form relation cannot contain any transitive dependencies.
- A relation R having functional dependency $A \rightarrow B$ is in 3NF if either of the conditions.
- 3NF non-prime attributes are only allowed to be functionally dependent on Super key of relation.
- A is a superkey.
- B is prime attribute, that is B is the part of candidate key.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=xoTyrdT9SZI>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 288 - 298)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems /19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : III -TRANSACTION PROCESSING

Date of Lecture: 07.09.2020

Topic of Lecture: Transaction Processing , Properties of Transactions

Introduction :

- Any logical calculation done in a consistent mode in a database is known as a transaction.
- A transaction is a sequence of READ and WRITE action that are grouped together to form a database.

Prerequisite knowledge for Complete understanding and learning of Topic:

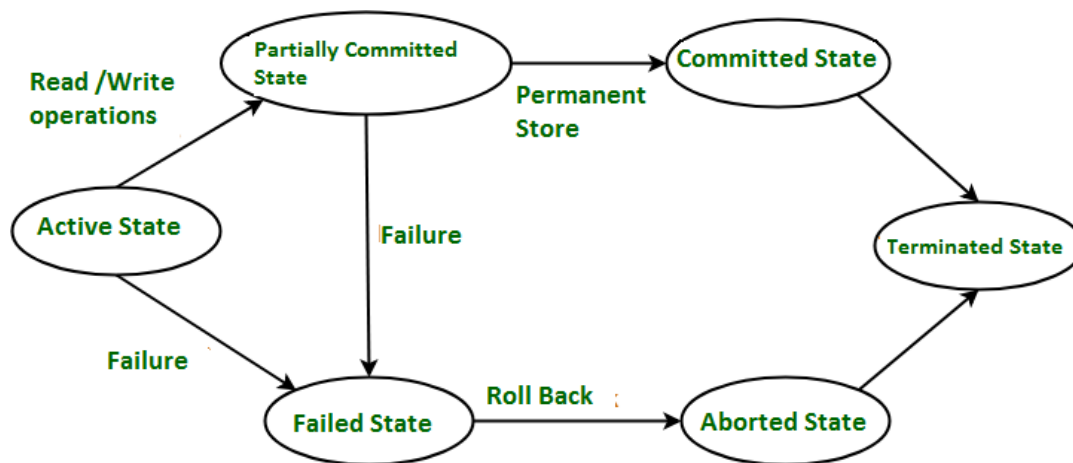
- DBMS
- ACID Properties
- Grouping

Detailed content of the Lecture:

Transaction Processing , Properties of Transactions:

Transaction Processing:

- Transaction in DBMS is a set of logically related operations. Transaction states in DBMS are the states through which a transaction goes throughout its lifetime.
- **Active :** The instructions of the transaction are running then the transaction is in active state.
- **Commit:** If it executes all its operations successfully. In this state, all the effects are now permanently saved on the database system.
- **Abort:** Abort a transaction, all database modifications performed under the protection of the transaction are discarded, and all locks currently held by the transaction are released
- **Partially Committed:** Transaction executes its final operation, it is said to be in a partially committed state.
- **Failed:** A transaction is said to be in a failed state if any of the checks made by the database recovery system fails. A failed transaction can no longer proceed further.
- **Terminate:** Any roll-back or the transaction comes from the “committed state”, then the system is consistent and ready for new transaction and the old transaction is terminated.



Transaction States in DBMS

Properties of Transactions:

- In the context of transaction processing, the acronym ACID refers to the four key.

Properties of a transaction:

- **Atomicity:** This property states that a transaction must be treated as an atomic unit, that is, either all of its operations are executed or none..
- **Consistency :** Data is in a consistent state when a transaction starts and when it ends.
- **Isolation :** The intermediate state of a transaction is invisible to other transactions. As a result, transactions that run concurrently appear to be serialized.
- **Durability :** After a transaction successfully completes, changes to data persist and are not undone, even in the event of a system failure.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=8RqtFpHkUsQ>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 609 - 616)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / 1

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs.R.Pavithra

Unit : III -TRANSACTION PROCESSING

Date of Lecture: 08.09.2020

Topic of Lecture: Serializability

Introduction :

- Transactions in a database correspond to the set of instructions that are executed to achieve a target.
- A given non-serial schedule of 'n' Transactions is said to be serializable if there exists some kind of equivalent serial schedule to the same 'n' transactions.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Transaction
- Database
- Records

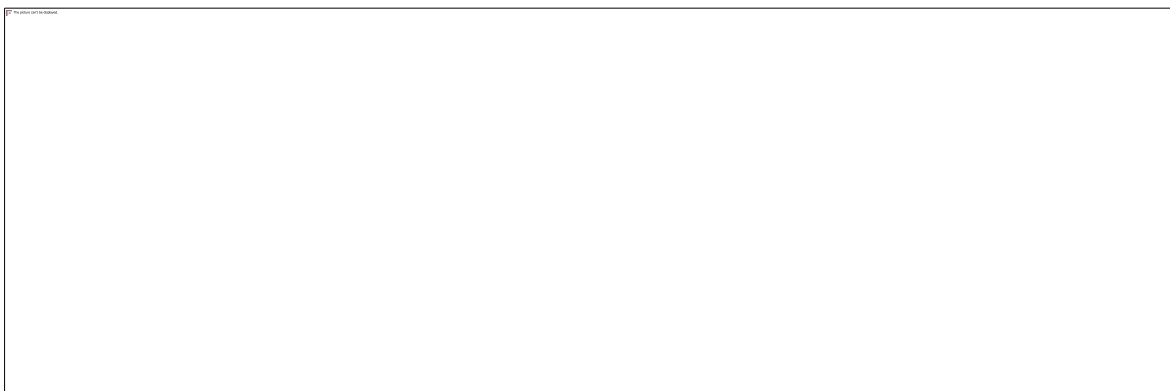
Detailed content of the Lecture:

Serializability:

- when a set of transactions are scheduled non-serially, they are interleaved leading to the problem of concurrency within the database.
- Non-serial schedules do not wait for one transaction to complete for the other one to begin.
- Serializability in DBMS decides if an interleaved non-serial schedule is serializable or not.

Example of Serializability

Consider 2 schedules, Schedule1 and Schedule2:



- Schedule1 is a serial schedule consisting of Transaction1 and Transaction2 wherein the operations on data item A (A1 and A2) are performed first and later the operations on data item B (B1 and B2) are carried out serially.

Types of Serializability:

- Conflict Serializability
- View Serializability.

Conflict Serializability:

- A conflict operation such as Read-Write or Write-Read or Write-Write is implemented on the same data item at the same time within different transactions then the schedule holding such transactions is said to be a conflict schedule.
- The conflict operations are to be implemented on the same data item.
- The conflict operations (RW, WR, WW) must take place within different transactions.
- At least one of the conflict operations must be the write operation.
- Two Read operations will not create any conflict.



View Serializability:

- Two schedules (one being serial schedule and another being non-serial) are said to be view serializable if they satisfy the rules for being view equivalent to one another.
- Initial values of the data items involved within a schedule must be the same.
- Final values of the data items involved within a schedule must be the same.
- The number of WR operations performed must be equivalent for the schedules involved.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=s8QIJol1G6w>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 620 - 625)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : III -TRANSACTION PROCESSING

Date of Lecture: 10.09.2020

Topic of Lecture: Transaction support in SQL , Locking Techniques

Introduction :

- An introduction to transaction support in SQL there are many more details, and the newer standards have more commands for transaction processing.
- Locking protocols are used in database management systems as a means of concurrency control. Multiple transactions may request a lock on a data item simultaneously.

Prerequisite knowledge for Complete understanding and learning of Topic:

- SQL
- Transaction
- TCL

Detailed content of the Lecture:

Transaction support in SQL:

- The basic definition of an SQL transaction is similar to our already defined concept of a transaction.
- That is, it is a logical unit of work and is guaranteed to be atomic.
- A single SQL statement is always considered to be atomic—either it completes execution without an error or it fails and leaves the database unchanged.

Access Mode:

- The access mode can be specified as READ ONLY or READ WRITE.
- The default is READ WRITE, unless the isolation level of READ UNCOMMITTED is specified (see below), in which case READ ONLY is assumed.
- A mode of READ WRITE allows select, update, insert, delete, and create commands to be executed.
- A mode of READ ONLY, as the name implies, is simply for data retrieval.

Diagnostic Area Size:

- The diagnostic area size option, DIAGNOSTIC SIZE n , specifies an integer value n , which indicates the number of conditions that can be held simultaneously in the diagnostic area.
- These conditions supply feedback information (errors or exceptions) to the user or program on the n most recently executed SQL statement.

Isolation Level:

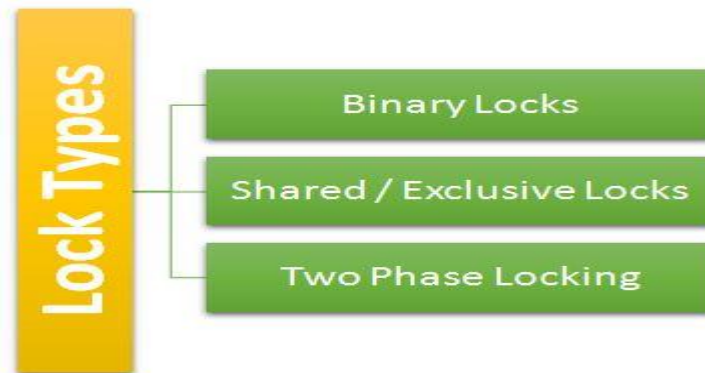
- The value for <isolation> can be READ UNCOMMITTED, READ

COMMITTED, REPEATABLE READ, or SERIALIZABLE.

- Dirty read
- Phantoms
- Nonrepeatable read.

Locking Techniques:

- we require a mechanism to manage the locking requests made by transactions. Such mechanism is called as Lock Manager.



Binary Locks:

- A binary lock has two states or values associated with each data item. These values are:
 - Locked – 1
 - Unlocked – 0
- If a data item is locked, then it cannot be accessed by other transactions i.e., other transactions are forced to wait until the lock is released by the previous transaction.
- But, if a data item is in the unlocked state, then, it can be accessed by any transaction and on access the lock value is set to locked state.

Exclusive Locks:

- Exclusive locks on the other hand are applied on the transactions which request a write operation on the data item.
- The transaction which is modifying the data item requests an exclusive lock on the data item and hence any other transaction which needs access to the data item has to wait until the lock applied by the previous transaction has been released by it.

Shared Locks :

- Shared locks are applied to a data item when the transaction requests a read operation on the data item. A shared lock will allow multiple transactions to only read the data item concurrently.
- As these locks are applied on read operation, they will not compromise on the consistency of the database.

Two Phase Locking :

- The Two Phase Locking Techniques guarantee Serializability in DBMS.
- A transaction is said to follow Two Phase Locking Protocol if all locking operations in the transaction precede the first unlock operation.

Locks are applied in two phases:

- Growing Phase
- Shrinking Phase

Growing Phase :

- This phase is also known as the first phase or the expanding phase. It is in this phase that the transaction acquires all the locks needed by it but it cannot release any locks here.

Shrinking Phase :

- This phase is also known as the second phase or the contracting phase. Here a transaction is not allowed to acquire any new locks but it can release the existing locks it holds. The Two Phase Locking Protocol helps solve problems of lost update, inconsistent analysis or dirty read too.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=FRs6tDVk-FY>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 635 - 642)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : III - TRANSACTION PROCESSING

Date of Lecture:21.09.2020

Topic of Lecture: Time Stamp ordering

Introduction :

- The Timestamp Ordering Protocol is used to order the transactions based on their Timestamps.
- To determine the timestamp of the transaction, this protocol uses system time or logical counter.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Transaction
- Database
- Protocol

Detailed content of the Lecture:

Time Stamp ordering:

- This clock is incremented when a transaction is submitted at that site and updated whenever the site receives a message with a higher clock value.
- Each transaction is assigned a unique timestamp and conflicting actions are executed in order of the timestamp of their transactions.

Advantages:

- Schedules are serializable just like 2PL protocols
- No waiting for the transaction, which eliminates the possibility of deadlocks!

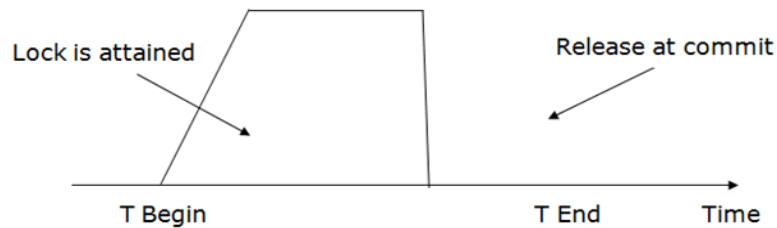
Disadvantages:

Starvation is possible if the same transaction is restarted and continually aborted

Strict Two-phase locking (Strict-2PL)

- The first phase of Strict-2PL is similar to 2PL. In the first phase, after acquiring all the locks, the transaction continues to execute normally.
- The only difference between 2PL and strict 2PL is that Strict-2PL does not release a lock after using it.
- Strict-2PL waits until the whole transaction to commit, and then it releases all the locks at a time.

- Strict-2PL protocol does not have shrinking phase of lock release.



Basic Timestamp works as follows:

ordering protocol

- Check the following condition whenever a transaction T_i issues a **Read (X)** operation:
 - If $W_TS(X) > TS(T_i)$ then the operation is rejected.
 - If $W_TS(X) \leq TS(T_i)$ then the operation is executed.
- Timestamps of all the data items are updated.
- 2. Check the following condition whenever a transaction T_i issues a **Write(X)** operation:
 - If $TS(T_i) < R_TS(X)$ then the operation is rejected.
 - If $TS(T_i) < W_TS(X)$ then the operation is rejected and T_i is rolled back otherwise the operation is executed.

Thomas write Rule:

- Thomas Write Rule provides the guarantee of serializability order for the protocol. It improves the Basic Timestamp Ordering Algorithm.
- The basic Thomas write rules are as follows:
 - If $TS(T) < R_TS(X)$ then transaction T is aborted and rolled back, and operation is rejected.
 - If $TS(T) < W_TS(X)$ then don't execute the $W_item(X)$ operation of the transaction and continue processing.
 - If neither condition 1 nor condition 2 occurs, then allowed to execute the WRITE operation by transaction T_i and set $W_TS(X)$ to $TS(T)$.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=27NtGV1vNoY>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 648 - 651)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : III - TRANSACTION PROCESSING

Date of Lecture:22.09.2020

Topic of Lecture:Validation Techniques , Granularity of Data Items

Introduction :

- Validation techniques in machine learning are used to get the error rate of the ML model, which can be considered as close to the true error rate of the population.
- It is the size of the data item allowed to lock. Now Multiple Granularity means hierarchically breaking up the database into blocks that can be locked and can be tracked needs what needs to lock and in what fashion.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Primary Key
- Locking Techniques
- Concurrency

Detailed content of the Lecture:

Validation Techniques:

- Validation Based Protocol is also called Optimistic Concurrency Control Technique.
- This protocol is used in DBMS (Database Management System) for avoiding concurrency in transactions.
- It is called optimistic because of the assumption it makes, i.e. very less interference occurs, therefore, there is no need for checking while the transaction is executed.
- Until the transaction end is reached updates in the transaction are not applied directly to the database.
- All updates are applied to local copies of data items kept for the transaction. At the end of transaction execution, while execution of the transaction, a validation phase checks whether any of transaction updates violate serializability.
- If there is no violation of serializability the transaction is committed and the database is updated; or else, the transaction is updated and then restarted.
- **Read Phase:** Values of committed data items from the database can be read by a transaction. Updates are only applied to local data versions.
- **Validation Phase:** Checking is performed to make sure that there is no violation of serializability when the transaction updates are applied to the database.
- **Write Phase:** On the success of the validation phase, the transaction updates are applied to the database, otherwise, the updates are discarded and the transaction is slowed down.

Granularity of Data Items :

- Granularity is the level of detail at which data are stored in a database. When the same data are represented in multiple databases, the granularity may differ.
- **Intention-Shared (IS):** explicit locking at a lower level of the tree but only with shared locks.
- **Intention-Exclusive (IX):** explicit locking at a lower level with exclusive or shared locks.
- **Shared & Intention-Exclusive (SIX):** the subtree rooted by that node is locked explicitly in shared mode and explicit locking is being done at a lower level with exclusive mode locks.

Database level locking:

Database level locks, the entire database is locked – which means that only one database session can apply any updates to the database. This type of lock is not often used, because it obviously prevents all users except one from updating anything in the database.

Table-Level Locking: :

Table-Level Locking. MySQL uses table-level locking for MyISAM , MEMORY , and MERGE tables, permitting only one session to update those tables at a time. This locking level makes these storage engines more suitable for read-only, read-mostly, or single-user applications.

Page level locking:

All the data on a specific page are locked. A page is a common unit of storage in computer systems and is used by all types of DBMS.

Row-level locking :

Row-level locking means that only the row that is accessed by an application will be locked. Hence, all other rows that belong to the same page are free and can be used by other applications. The Database Engine can also lock the page on which the row that has to be locked is stored.

Attributes level locking:

At attribute level locking, particular attribute (or field) is locked. Attribute level locking allows concurrent transactions to access the same row, as long as they require the use of different attributes within the row.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=q_-fhQXOQbg

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (**Page No : 651 - 656**)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : III - TRANSACTION PROCESSING

Date of Lecture: 23.09.2020

Topic of Lecture: Recovery concepts

Introduction :

- Database recovery is the process of restoring the database to a correct (consistent) state in the event of a failure.
- In other words, it is the process of restoring the database to the most recent consistent state that existed shortly before the time of system failure.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Database
- System Failure
- Files
- Tables

Detailed content of the Lecture:

Recovery concepts:

- Recovery is the single most important part of any training or exercise program.
- Recovery allows for improved performance, permits time for our body to heal itself in preparation for the next training load, and decreases the risk of potential injury.

Transaction Recovery:

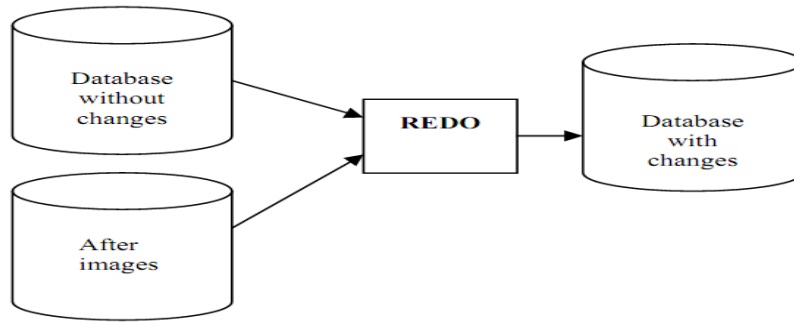
- Deferred update
- Immediate update
- Caching/Buffering
- Shadow paging

Types of Database Recovery:

- Forwarded Recovery (REDO).
- Backwarded Recovery (UNDO).

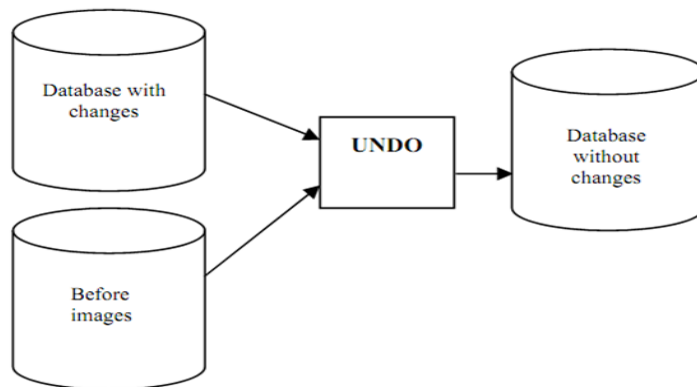
Forwarded Recovery (REDO):

- Forward Recovery (Redo): In this system the committed changes made by a transaction are reapplied to a previous copy of the database.



Backward Recovery(UNDO):

- Backward Recovery (UNDO): In this system the uncommitted changes made by a transaction to a database are undone.
- Instead the system is reset to the last consistent state of database that is free from any errors.



Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=HnVo3_iH76w

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 688 - 697)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : III -TRANSACTION PROCESSING

Date of Lecture: 24.09.2020

Topic of Lecture: Shadow Paging

Introduction :

- Shadow Paging is recovery technique that is used to recover database.
- In this recovery technique, database is considered as made up of fixed size of logical units of storage which are referred as pages.

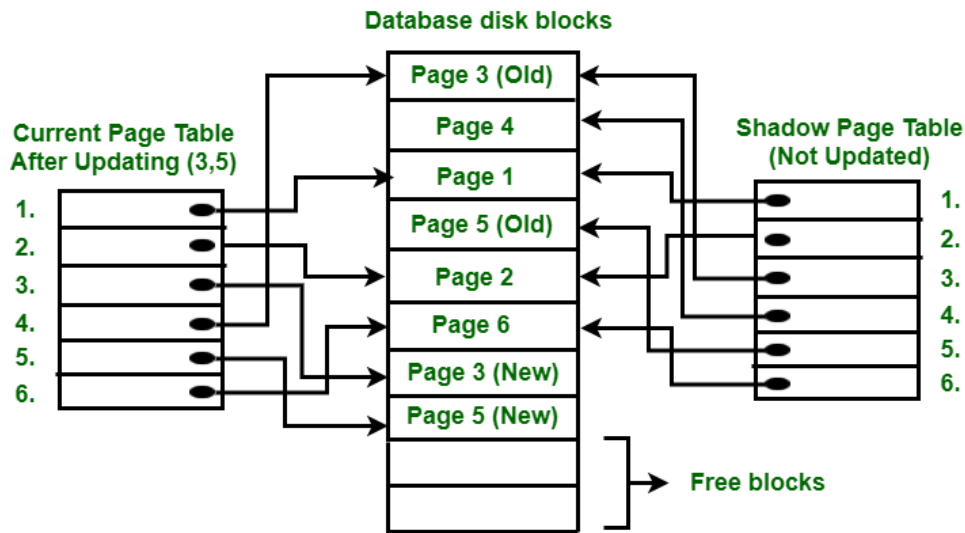
Prerequisite knowledge for Complete understanding and learning of Topic:

- Database
- Page Table
- Free block

Detailed content of the Lecture:

Shadow Paging :

- Pages are mapped into physical blocks of storage, with help of the page table which allow one entry for each logical page of database.
- This method uses two page tables named current page table and shadow page table.
- The entries which are present in current page table are used to point to most recent database pages on disk.
- Another table i.e., Shadow page table is used when the transaction starts which is copying current page table.
- After this, shadow page table gets saved on disk and current page table is going to be used for transaction.
- This technique is also known as Cut-of-Place updating.
- Firstly, search start for available free block in disk blocks.
- After finding free block, it copies page 3 to free block which is represented by Page 3 (New).
- Now current page table points to Page 3 (New) on disk but shadow page table points to old page 3 because it is not modified.
- The changes are now propagated to Page 3 (New) which is pointed by current page table.



Advantages :

- This method require fewer disk accesses to perform operation.
- In this method, recovery from crash is inexpensive and quite fast.
- There is no need of operations like- Undo and Redo.

Disadvantages :

- Due to location change on disk due to update database it is quite difficult to keep related pages in database closer on disk.
- During commit operation, changed blocks are going to be pointed by shadow page table which have to be returned to collection of free blocks otherwise they become accessible.
- The commit of single transaction requires multiple blocks which decreases execution speed.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=495q-pDJ9zE>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 699 - 702)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems /19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : III - TRANSACTION PROCESSING

Date of Lecture: 25.09.2020

Topic of Lecture: Concurrency control , Log Based Recovery

Introduction :

- Concurrency Control in Database Management System is a procedure of managing simultaneous operations without conflicting with each other.
- The log is a sequence of records. Log of each transaction is maintained in some stable storage so that if any failure occurs, then it can be recovered from there.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Functions
- Data Integrity
- Data Modification

Detailed content of the Lecture:

Concurrency control :

- Concurrency Control in Database Management System is a procedure of managing simultaneous operations without conflicting with each other.
- It ensures that Database transactions are performed concurrently and accurately to produce correct results without violating data integrity of the respective Database.

Types of Concurrency control :

- Concurrency control With Locking.
- Concurrency control Without Locking.

Concurrency control With Locking:

- Locks are an integral part to maintain concurrency control in DBMS.
- A transaction in any system implementing lock based concurrency control cannot read or write a statement until it has obtained the required locks.
- **Binary Locks** - These can only be in one of two states, locked or unlocked.
- **Shared/Exclusive Locks** - Shared locks are acquired when only read operation is to be performed. Shared locks can be shared between multiple transactions as there is no data being altered. Exclusive locks are used when write operation is performed. Only the transaction holding the exclusive lock is allowed to make changes to the data value.

Two Phase Locking Protocol:

- Two Phase Locking Protocol also known as 2PL protocol is a method of concurrency control in

DBMS that ensures serializability by applying a lock to the transaction data which blocks other transactions to access the same data simultaneously.

- **Growing Phase:** In this phase transaction may obtain locks but may not release any locks.
- **Shrinking Phase:** In this phase, a transaction may release locks but not obtain any new lock

Concurrency control Without Locking:

- The field of databases, non-lock concurrency control is a concurrency control method used in relational databases without using locking.
- There are several non-lock concurrency control methods, which involve the use of timestamps on transaction to determine transaction priority:
- Optimistic concurrency control
- Timestamp-based concurrency control
- Multiversion concurrency control

Log Based Recovery:

The database can be modified using two approaches –

- **Deferred Modification Technique:** If the transaction does not modify the database until it has partially committed, it is said to use deferred modification technique.
- **Immediate Modification Technique:** If database modification occur while transaction is still active, it is said to use immediate modification technique.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=VR7tpafzeW0>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 635 - 648)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : III - TRANSACTION PROCESSING

Date of Lecture: 28.09.2020

Topic of Lecture: Deadlock Handling, Insert and Delete Operations.

Introduction :

- In a database, a deadlock is an unwanted situation in which two or more transactions are waiting indefinitely for one another to give up locks.
- Insert is a widely-used command in the Structured Query Language (SQL) data manipulation language (DML) used by SQL Server and Oracle relational databases.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Data Failure
- SQL Operations
- Relational Database

Detailed content of the Lecture:

Deadlock Handling:

- Deadlock is said to be one of the most feared complications in DBMS as no task ever gets finished and is in waiting state forever.

Deadlock Avoidance

- When a database is stuck in a deadlock state, then it is better to avoid the database rather than aborting or restating the database. This is a waste of time and resource.
- Deadlock avoidance mechanism is used to detect any deadlock situation in advance. A method like "wait for graph" is used for detecting the deadlock situation but this method is suitable only for the smaller database. For the larger database, deadlock prevention method can be used.
- **Deadlock Detection**
- In a database, when a transaction waits indefinitely to obtain a lock, then the DBMS should detect whether the transaction is involved in a deadlock or not. The lock manager maintains a Wait for the graph to detect the deadlock cycle in the database.
- **Deadlock Prevention**
- Deadlock prevention method is suitable for a large database. If the resources are allocated in

such a way that deadlock never occurs, then the deadlock can be prevented.

- The Database management system analyzes the operations of the transaction whether they can create a deadlock situation or not. If they do, then the DBMS never allowed that transaction to be executed.

Insert and Delete Operations:

- The execute simple INSERT, UPDATE, SELECT, and DELETE statements qualified by a WHERE clause on the target table or view. These operations are also called data manipulation language (DML) operations.
- The Insert operation is called Enqueue, and adds an element to the end of the list; the Dequeue operation selects and removes its first element.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=YeXS1JWIA4Q>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 659 - 667)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : IV -FILES AND INDEXING

Date of Lecture: 01.10.2020

Topic of Lecture: File operations

Introduction :

- File organisation refers to the logical structuring of the records as determined by the way in which they are accessed.
- File organization is a way of arranging the records in a file when the file is stored on the disk.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Insert
- Delete
- Data Modify
- Records

Detailed content of the Lecture:

File operations:

- File organisation refers to the logical structuring of the records as determined by the way in which they are accessed.
- File organization is a way of arranging the records in a file when the file is stored on the disk.
- A file can be created , destroyed and have records inserted into and delete from it.

File Operations:

- **Open:** This operation prepare the file for reading or writing.
- **Reset:** Sets the file pointer of an open file to the beginning of the file.
- **Find:** Searches for the first records that satisfies a search condition.
- **Read:** Copies the current record from the buffer to a program variable in the user program.
- **Find Next:** Searches for the next record in the file that satisfies the search condition. Transfers the block containing that record into a main memory buffer.
- **Delete:** Delete the current record and updates the file on disk to reflect the deletion.
- **Modify:** Modifies some field values for the current record and updates the file on disk to reflect the modification.
- **Insert:** Inserts a new record in the file by locating the block where the record is to inserted, transferring that block into a main memory buffer.
- **Close:** Completes the file access by releasing the buffers and performing any other needed cleanup operations.

- **Scan** : If the file has just been opened or reset, scan returns the first record; otherwise it returns the next record.
- **Find All**: Locates all the records in the file that satisfy a search condition.
- **Find Ordered**: Retrieves all the records in the file in some specified order.
- **Re-Organise**: Starts the re-organisation process.

Fields and Records:

- A record is an entity composed of data items or fields in a file.
- Each data item is formed of one or more bytes and corresponds to a particular field of the record.
- Fixed-length record.
- Variable-length record.
- **Fixed-length record**: In a file with fixed-length records, all records on the page are of the same slot length. Every records in the exactly the same size.
- **Variable-length record**: All records on the page are not of the same length. The records is different records in the file have different sizes.

There are two way of implementation:

- Byte-string Representation
- Fixed-length Representation

Byte-string Representation:

- Different techniques are used to implement variable-length records operation in a file.
- Byte-string representation in one of the simplest techniques of implementing variable-length operation.

Disadvantages:

- It is very difficult to re-use empty slot space occupied formally by deleted records.
- A large number of small fragments of disk storage are wasted.

Fixed-length Representation:

- It is another techniques to implement variable-length records operation in a file. It is one or more fixed-length records are used to represent variable-length record.
- **Reserved Space** : A fixed-length record of the size equal to that of maximum record length in a file is used.
- **List Representation**: A list representation a list of fixed-length records, chained together by pointers, is used to implement variable-length record.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=_ChkXLOabZw

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 464 - 467)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / 1

Course Name with Code : Relational Database Management System / 19CAB03

Course Faculty : Mrs.R.Pavithra

Unit : IV -FILES AND INDEXING

Date of Lecture: 03.10.2020

Topic of Lecture: File Organisation Techniques

Introduction :

- A database consist of a huge amount of data. The data is grouped within a table in RDBMS, and each table have related records.
- A user can see that the data is stored in form of tables, but in actual this huge amount of data is stored in physical memory in form of files.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Physical Memory
- Logical Data
- Sequential Data

Detailed content of the Lecture:

File Organisation Techniques :

- File Organization refers to the logical relationships among various records that constitute the file, particularly with respect to the means of identification and access to any specific record.
- In simple terms, Storing the files in certain order is called file Organization.
- File Structure refers to the format of the label and data blocks and of any logical control record.
- Heap Access File.
- Sequential Access File.
- Index Sequential Access File.
- Direct Access File.

Heap Access File:

- The simplex form of file organization may be termed the pile .
- A heap or pile file records are collected in the order they arrive.
- Pointer link the block used in a heap. A new record is to be inserted, it is placed in the last block if there is a space.

Advantages:

- This is a simple file organisation method.

- Insertion is somehow efficient.
- Good for bulk-loading data into a table.
- Best if file scans are common or insertions are frequent.

Disadvantages:

- Retrieval requires a linear search and is inefficient.
- Deletion can result in unused space/need for re-organisation.

Sequential Access File:

- The most basic way to organise the collection of records in a file is to use sequential organisation.
- Records of the file are stored in sequence by the primary key field values.

Advantages:

- It is fast and efficient when dealing with large volumes of data that need to be processed periodically.
- Simple to implement.
- Requires very low software support.
- Efficiency of blocking is good.

Disadvantages:

- Requires that all new transactions be started into the proper sequence for sequential access processing.
- Locating, storing, modifying, deleting, or adding records in the file require rearranging the file.
- This method is too slow to handle applications requiring immediate updating or responses.

Index Sequential Access File:

- The index sequential file organization is a hybrid organization which uses elements of indexed and sequential file organizations

Advantages:

- This organisation also provide quick access to records in a relatively efficient way.
- Records can be inserted or updated in the middle of the files.

Disadvantages:

- Indexed file organisation is less efficient in the storage space than some other file organisation.
- Processing is slow.
- It requires relatively expensive hardware and software resources.

Direct Access File:

- Files records can be read in any order are called direct access or random access files.
- Random access files are basically a collection of records stored on a disk and these records.

Relative Addressed Direct File Organisation:

- In this type of file organisation the relative address of a desired record is derived from the base address.

Hashed Addressed Direct File Organisation:

- The keys of records in the hashed addressed direct file organisation.

Advantages:

- It given fastest retrieval of records.
- It gives efficient use of memory.
- Supports fast storage devices.

Disadvantages:

- Less efficient use of storage spaces.
- Direct file organisation is device dependent.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=y-LeibeXAog>

Important Books/Journals for further learning including the page nos.:

Database System Concepts – Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hill, Sixth Edition, 2010 (Page No : 468- 474)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : IV -FILES AND INDEXING

Date of Lecture: 07.10.2020

Topic of Lecture: Hashing Techniques

Introduction :

- The usual method of direct mapping is by performing some arithmetic manipulation of the key value. This process is called hashing.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Indexes
- Keys
- Records
- Arithmetic Operations

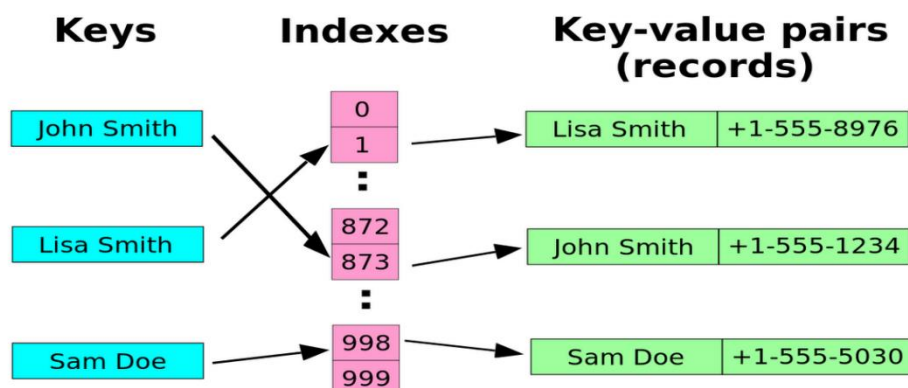
Detailed content of the Lecture:

Hashing Techniques:

- The usual method of direct mapping is by performing some arithmetic manipulation of the key value. This process is called hashing.

Hash Table:

- A hash table , or a hash map, is a data structure that associates keys with values.



Hashing Function:

- A hash function is any well-defined procedure or mathematical function which converts a large, possibly variable-sized amount of data into a small datum, usually a single integer that may serve as an index into a array.

Properties of good hash Function:

- Low Cost
- Determinism
- Uniformity
- Variable Range
- Continuity

Hashing Techniques:

- Hashing is a techniques by which key field is converted into address of physical location or record by using any function, known as hash function.

Types of Hashing Techniques:

- Division Hash Function.
- Multiplication Hash Function.
- Mid-Square Hash Function.
- Folding Hash Function.

Division Method:

- Simple choice for a hash function is obtained by using the modulo(mod) operator. The identifier x is divided by some number m and the remainder is used as the hash address of K . $h(k)=k \bmod m$

Multipulation Method:

- It multiplies all the individual digits in the key together, and takes the remainder after dividing the resulting number by the table size. $h(X) = (a*b*c*d*....)\bmod m$

Mid-Square Method:

- The middle of square function is computed by squaring the identifier and then using an appropriate number of bits from the middle of the square to obtain the bucket address.

Folding Method:

- In the first step, the key value k is divided into number of parts, k_1 ,
- k_2, k_3, \dots, k_r , where each part has the same number of digits except the last part which can have less digits.

Shift folding:

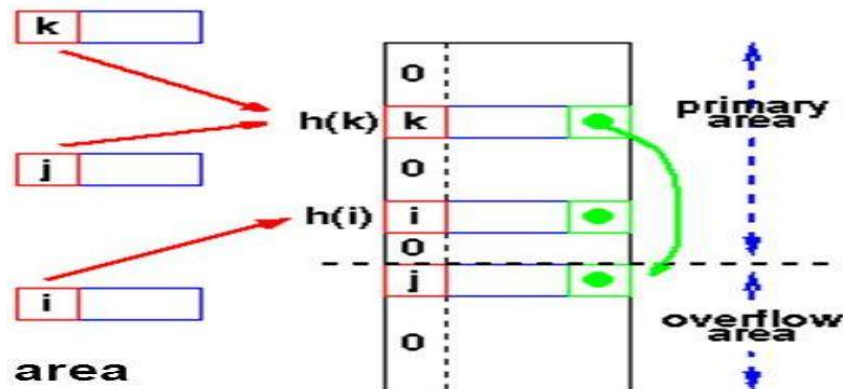
- The other method of adding the parts is folding at the boundaries. The digits falling into the same position are added together.
-

Collision in Hashing:

- A key are inserted in the table, it is possible that two keys may hash to the same table slot.
- The hash function distributes the elements uniformly over the table the number of collisions cannot be too large on the average.

Overflow Area:

- The general way are keeping sub sequence items within the table and computing possible location.
- Another schema will divide the per-allocated table into two sections the primary area to which keys are mapped and an area for collisions.



Collision Resolution Techniques:

- Opening Hashing or Chaining.
- Closed Hashing or Open Addressing
- Rehashing.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=zeMa9sg-VJM>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No :506 - 509)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems /19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : IV -FILES AND INDEXING

Date of Lecture:09.10.2020

Topic of Lecture: Indexing

Introduction :

- Indexing is a way of providing a fast access path to the values of a column.
- An index is a small table having only two columns.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Records
- Hashing
- Files Organizations

Detailed content of the Lecture:

Indexing:

- Indexing is a way of providing a fast access path to the values of a column.
- An index is a small table having only two columns.

Index Sequential Access File:

- The index sequential file organization is a hybrid organization which uses elements of indexed and sequential file organizations .

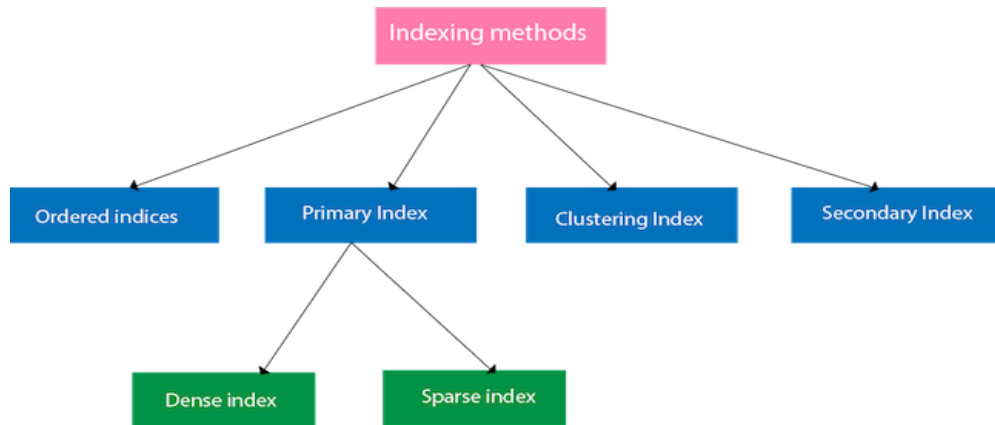
Advantages:

- This organization also provide quick access to records in a relatively efficient way.c
- Records can be inserted or updated in the middle of the files.

Disadvantages:

- Indexed file organization is less efficient in the storage space than some other file organization.
- Processing is slow.
- It requires relatively expensive hardware and software resources.

Indexing Methods



Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=E--yzX05_k8

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 481 - 483)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : IV -FILES AND INDEXING

Date of Lecture:12.10.2020

Topic of Lecture:Single level

Introduction :

- Indexing is a small table which is consist of two columns.
- Indexing is a way to optimize the performance of a database by minimizing the number of disk accesses required when a query is processed.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Records
- Files
- Database
- Indices.

Detailed content of the Lecture:

Single level :

- A single-level index is an auxiliary file that makes it more efficient to search for a record in the data file.

Ordered indices:

- The indices are usually sorted to make searching faster. The indices which are sorted are known as ordered indices.

Example: Suppose we have an employee table with thousands of record and each of which is 10 bytes long. If their IDs start with 1, 2, 3....and so on and we have to search student with ID-543.

- In the case of a database with no index, we have to search the disk block from starting till it reaches 543. The DBMS will read the record after reading $543 \times 10 = 5430$ bytes.
- In the case of an index, we will search using indexes and the DBMS will read the record after reading $542 \times 2 = 1084$ bytes which are very less compared to the previous case.

Primary Index:

- If the index is created on the basis of the primary key of the table, then it is known as primary

indexing. These primary keys are unique to each record and contain 1:1 relation between the records.

- As primary keys are stored in sorted order, the performance of the searching operation is quite efficient.
- The primary index can be classified into two types: Dense index and Sparse index.

Dense index:

- The dense index contains an index record for every search key value in the data file. It makes searching faster.
- In this, the number of records in the index table is same as the number of records in the main table.
- It needs more space to store index record itself. The index records have the search key and a pointer to the actual record on the disk.

Sparse index:

- In the data file, index record appears only for a few items. Each item points to a block.
- In this, instead of pointing to each record in the main table, the index points to the records in the main table in a gap.

Clustering Index:

- A clustered index can be defined as an ordered data file. Sometimes the index is created on non-primary key columns which may not be unique for each record.
- In this case, to identify the record faster, we will group two or more columns to get the unique value and create index out of them. This method is called a clustering index.
- The records which have similar characteristics are grouped, and indexes are created for these group.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=Ej4oUU2eaFk>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 482 - 487)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : IV -FILES AND INDEXING

Date of Lecture: 19.10.2020

Topic of Lecture: Multilevel Index

Introduction :

- Index records comprise search-key values and data pointers. Multilevel index is stored on the disk along with the actual database files. As the size of the database grows, so does the size of the indices

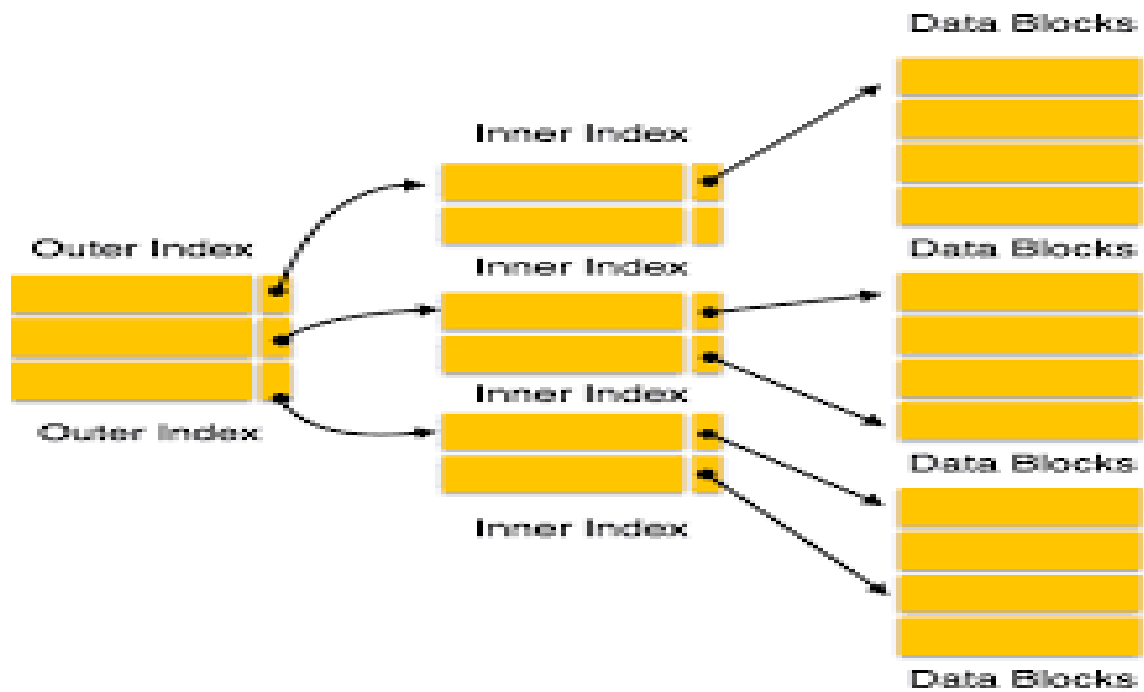
Prerequisite knowledge for Complete understanding and learning of Topic:

- Primary Index
- Pointer
- Database
- Index Records

Detailed content of the Lecture:

Multilevel Index:

- Multilevel Indexing in Database is created when a primary index does not fit in memory. In this type of indexing method, you can reduce the number of disk accesses to short any record and kept on a disk as a sequential file and create a sparse base on that file.
- Index records comprise search-key values and data pointers.
- Multilevel index is stored on the disk along with the actual database files.
- As the size of the database grows, so does the size of the indices.
- There is an immense need to keep the index records in the main memory so as to speed up the search operations.
- If single-level index is used, then a large size index cannot be kept in memory which leads to multiple disk accesses.



- Multi-level Index helps in breaking down the index into several smaller indices in order to make the outermost level so small that it can be saved in a single disk block, which can easily be accommodated anywhere in the main memory.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=Ua08uVgsk4k>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 502 - 506)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : IV -FILES AND INDEXING

Date of Lecture: 26.10.2020

Topic of Lecture: B+ tree

Introduction :

- A B+ tree consists of a root, internal nodes and leaves.
- A B+ tree can be viewed as a B-tree in which each node contains only keys (not key-value pairs), and to which an additional level is added at the bottom with linked leaves.

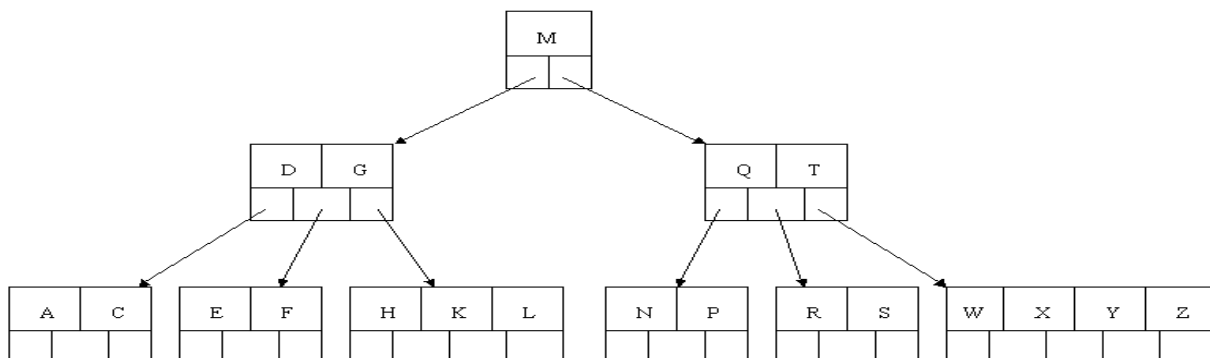
Prerequisite knowledge for Complete understanding and learning of Topic:

- Nodes
- Keys
- Internal Nodes
- Index

Detailed content of the Lecture:

B+ Tree:

- A B+ tree is a balanced binary search tree that follows a multi-level index format. The leaf nodes of a B+ tree denote actual data pointers.
- B+ tree ensures that all leaf nodes remain at the same height, thus balanced.
- Additionally, the leaf nodes are linked using a link list; therefore, a B+ tree can support random access as well as sequential access.



Structure of B+ Tree:

- Every leaf node is at equal distance from the root node. A B+ tree is of the order n where n is fixed for every B+ tree.

Internal nodes :

- Internal (non-leaf) nodes contain at least $\lceil n/2 \rceil$ pointers, except the root node.
- At most, an internal node can contain n pointers.

Leaf nodes :

- Leaf nodes contain at least $\lceil n/2 \rceil$ record pointers and $\lceil n/2 \rceil$ key values.
- At most, a leaf node can contain n record pointers and n key values.
- Every leaf node contains one block pointer P to point to next leaf node and forms a linked list.

B+ Tree Insertion:

B+ trees are filled from bottom and each entry is done at the leaf node.

If a leaf node overflows

Split node into two parts.

Partition at $i = \lfloor (m+1)/2 \rfloor$.

First i entries are stored in one node.

Rest of the entries ($i+1$ onwards) are moved to a new node.

i th key is duplicated at the parent of the leaf.

If a non-leaf node overflows

Split node into two parts.

Partition the node at $i = \lceil (m+1)/2 \rceil$.

Entries up to i are kept in one node.

Rest of the entries are moved to a new node.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=aZjYr87r1b8>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 489 - 502)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : IV -FILES AND INDEXING

Date of Lecture: 27.10.2020

Topic of Lecture:Static Hashing

Introduction :

- Static hashing is a simple form of hashing, where hashing is the use of mathematical functions to sort incoming data in a speedy and organized fashion.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Insertion
- Deletion
- Searching

Detailed content of the Lecture:

Static Hashing:

- Static hashing is a simple form of hashing, where hashing is the use of mathematical functions to sort incoming data in a speedy and organized fashion.
- Static hashing is a simple form of hashing often used for a temporary solution a better of hashing.

Types of Operations in Static Hashing:

- Searching.
- Insertion.
- Deletion.

Dynamic Hashing Approach:

- The problem of lengthy searching of overflow buckets, is sloved by dynamic hashing.
- The dynamic hashing the size of the directory grows with the number of collision to accommodate new records and avoid long overflow page chains.

Two types of dynamic hashing approaches:

- Extendible hashing
- Linear hashing

Extendible hashing:

- It is a new access technique in which the user is guaranteed no more than two pages faults to locate the data associated with a given unique identifier or key.

Linear Hashing:

- Linear hashing is a dynamically updateable disk-based index structure which implements a hashing schema and which grows or shrinks one bucket at a time.

Extendible hashing versus Linear hashing:

Extendible hashing	Linear hashing
Extendible hashing use a bucket directory	Linear hashing dose not use a bucket directory
Over flown bucket is split	f overflow occurs it is not always the over flown bucket that is split
Number of bucket grows in a non linear fashion	Number of bucket grows in a linear fashion
Hashing function not changes dynamically	Hashing function change dynamically
There can be more than two hashing functions	There can be at most two hashing functions used by the schema

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=Q-RMoyaPjzE>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 506 - 513)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : IV -FILES AND INDEXING

Date of Lecture: 28.10.2020

Topic of Lecture: Indexes on Multiple Keys

Introduction :

- It is assumed that the primary or secondary keys on which files were accessed were single attributes.
- In many retrieve and updates requests,multiple attributes are involved.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Attributes
- Files
- Grid Files

Detailed content of the Lecture:

Indexes on Multiple Keys:

- It is assumed that the primary or secondary keys on which files were accessed were single attributes.
- In many retrieve and updates requests,multiple attributes are involved.

Indexes on multiple Attributes:

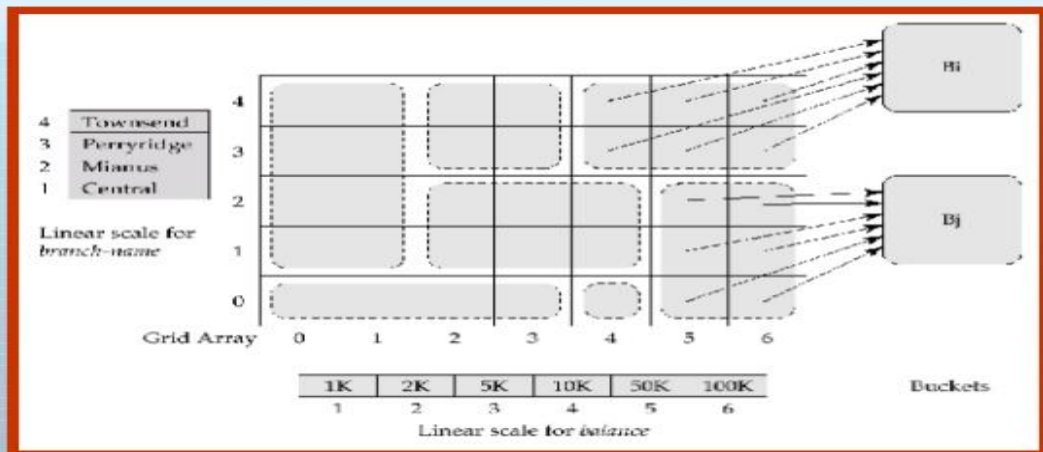
- One can create an index on a search key field that is a combination.

Grid Files:

- Another alternative is to organize the file as a grid.



Example Grid File for account



Advantages:

- No special computations are required
- Only the right records are retrieved
- Can also be used for single search key queries
- Easy to extend to queries on n search keys

Disadvantages:

- Imposes space overhead
- Performance overhead on insertion and deletion

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=LurtnV7UOFk>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No : 485 - 489)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : V -SPECIAL PURPOSE DATABASES

Date of Lecture: 02.11.2020

Topic of Lecture: OODBMS

Introduction :

- OODBMS stands for Object-Oriented Database Management System.
- It is a DBMS where data is represented in the form of objects, as used in object-oriented programming. OODB implements object-oriented concepts such as classes of objects, object identity, polymorphism, encapsulation, and inheritance.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Object , Database , Features, Object Oriented Programming

Detailed content of the Lecture:

OODBMS:

- An object database is a database management system in which information is represented in the form of objects as used in object-oriented programming.
- Object databases are different from relational databases which are table-oriented. Object-relational databases are a hybrid of both approaches.

Features of OODBMS:

- **Complexity:** OODBMS has the ability to represent the complex internal structure (of object) with multilevel complexity.
- **Inheritance:** Creating a new object from an existing object in such a way that new object inherits all characteristics of an existing object.
- **Encapsulation:** Encapsulation is one of the fundamentals of OOP (object-oriented programming). It refers to the bundling of data with the methods that operate on that data. Encapsulation is used to hide the values or state of a structured data object inside a class, preventing unauthorized parties' direct access to them.
- **Persistency:** A persistent database stores persistent data in the form of objects, or records that are durable when changing devices and software. Persistent data is stable and recoverable. Traditional relational database management systems (RDBMS) store persistent data in the form of records and tables.
- **Accessing Data:** In order to access data in a given database, you must first connect to the database. When you start SQL*Plus, you normally connect to your default Oracle database under the username and password you enter while starting.
- **Binding to Object:** Data binding, in the context of .NET, is the method by which controls on a user interface (UI) of a client application are configured to fetch from, or update data into, a data source, such as a database or XML document.
- **Saving Object:** A database object is any defined object in a database that is used to store or reference data. Anything which we make from create command is known as Database Object. It can be used to hold and manipulate the data. Some of the examples of database objects are : view, sequence, indexes, etc.
- **Queries:** In a relational database, which contains records or rows of information, the SQL SELECT statement query allows users to choose data and return it from a database to an application. The resulting query is stored in a result table
- **Indexing:** Indexing is used to optimize the performance of a database by minimizing the number of disk accesses required when a query is processed.
- **Transaction:** In a database management system, a transaction is a single unit of logic or work, sometimes made up of multiple operations. Any logical calculation done in a consistent mode in a database is known as a transaction.



Video Content / Details of website for further learning (if any):
<https://www.youtube.com/watch?v=AkoBIEGPLfg>

Important Books/Journals for further learning including the page nos.:
Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No :45-54)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



Course Name with Code : **Relational Database Management Systems / 19CAB03**

Course Faculty : **Mrs.R.Pavithra**

Unit : **V -SPECIAL PURPOSE DATABASES**

Date of Lecture: 03.11.2020

Topic of Lecture: Object-Based Databases

Introduction :

- Object oriented database systems are alternative to relational database and other database systems.
- In object oriented database, information is represented in the form of objects.
- Object oriented databases are exactly same as object oriented programming languages.

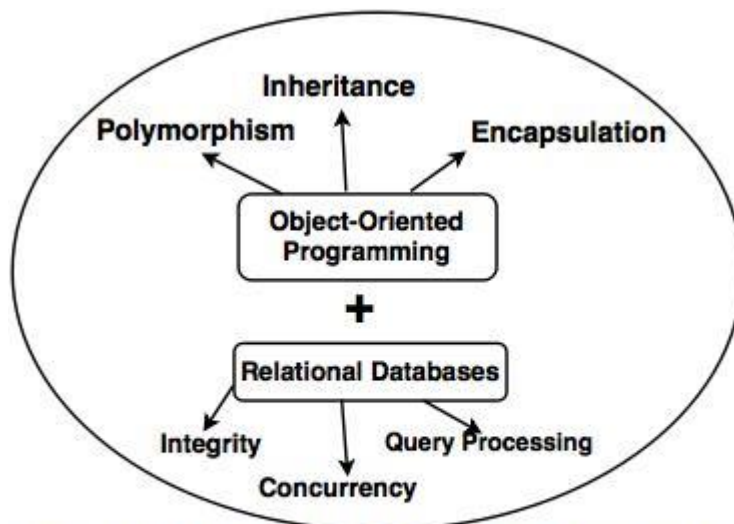
Prerequisite knowledge for Complete understanding and learning of Topic:

- Inheritance, Polymorphism, Encapsulation, Data Abstraction

Detailed content of the Lecture:

Object-Based Databases:

- Object oriented databases are exactly same as object oriented programming languages If we can combine the features of relational model (transaction, concurrency, recovery) to object oriented databases, the resultant model is called as object oriented database model.



(Object-Oriented database is product of OOP and RDB)

Features of **Object-Oriented database** OODBMS:

- In OODBMS, every entity is considered as object and represented in a table.
- Similar objects are classified to classes and subclasses and relationship between two object is maintained using concept of inverse reference.

Some of the features of OODBMS are as follows:

- 1. Complexity:** OODBMS has the ability to represent the complex internal structure (of object) with multilevel complexity.
- 2. Inheritance:** Creating a new object from an existing object in such a way that new object inherits all characteristics of an existing object.
- 3. Encapsulation:** It is an data hiding concept in OOPL which binds the data and functions together which can manipulate data and not visible to outside world.
- 4. Persistency :**OODBMS allows to create persistent object (Object remains in memory even after execution). This feature can automatically solve the problem of recovery and concurrency.

Challenges in ORDBMS implementation:

During the implementation of ORDBMS, various challenges arise which need to be resolved.

1. Storage and accessibility of data

- It is possible to define new types with new access to structures with the help of OODBMS. Hence, it is important that the system must store ADT and structured objects efficiently along with the provision of indexed access.

2. Query Processing

- Efficient flow of Query Processing and optimization is a difficult task.

3. Query Optimization

- New indexes and query processing techniques increase the options for query optimization. But, the challenge is that the optimizer must know to handle and use the query processing functionality properly.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=meWQLWq7QSE>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No :89-96)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

I / I

Course Name with Code : **Relational Database Management Systems / 19CAB03**

Course Faculty : **Mrs. R.Pavithra**

Unit : **V -SPECIAL PURPOSE DATABASES**

Date of Lecture: 04.11.2020

Topic of Lecture: OO Data Model - OO Languages

Introduction :

- The ODBMS which is an abbreviation for object-oriented database management system is the data model in which data is stored in form of objects, which are instances of classes.
- An object-oriented database management system represents information in the form of objects as used in object-oriented programming.

Prerequisite knowledge for Complete understanding and learning of Topic:

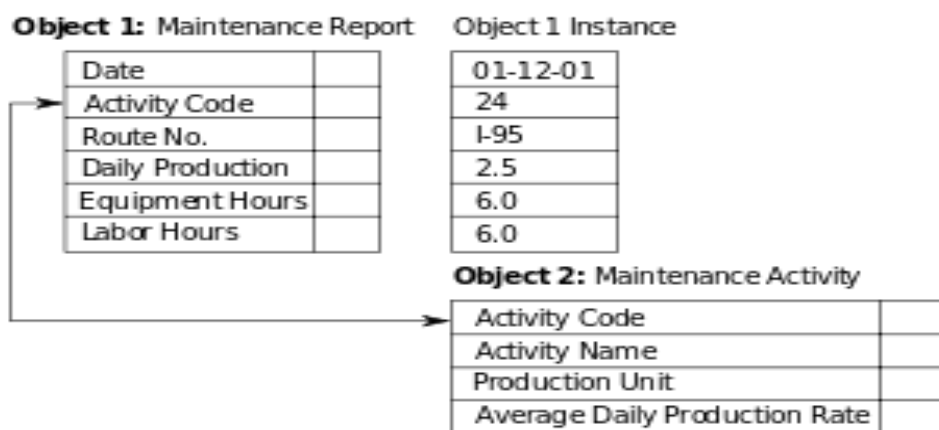
- Object-Oriented Programming
- Classes
- Objects
- DBMS

Detailed content of the Lecture:

OO Data Model:

- An object data model is a data model based on object-oriented programming, associating methods (procedures) with objects that can benefit from class hierarchies.
- An object-oriented data model is one that extends the individual program space into the world of persistent object management and shareability.

Object-Oriented Model



Advantages of Object-Oriented database:

- Capable to handle many different types of data types.
- Improves Productivity.
- Data Access.

Disadvantages of Object-Oriented database:

- Difficult to Maintain.
- Not suited for all Application.

OO Languages:

- An object-oriented database management system represents information in the form of objects as used in object-oriented programming.
- Certain object-oriented databases are designed to work with object-oriented programming languages such as Delphi, Python, Java, Perl, Objective C and Visual Basic . NET

Different Types of Object-Oriented Languages:

- **Simula** : Simula, short for "simulation language," was the first object-oriented programming language. In the late 1960s, the Norwegian developers of Simula, O. J. Dahl and Kristen Nygaard, were looking for a way for programs to simulate or model .
- **Small talk:** Smalltalk is an object-oriented, dynamically typed reflective programming language. Smalltalk was created as the language underpinning the "new world" of computing exemplified by "human-computer symbiosis".
- **C++:** This database was launched in 1989 and it is also one of the most popular relational database management systems (RDBMS) in the world. It is written in C and C++ and supports structured query language.
- **Java:** OOP concepts in Java are the main ideas behind Java's Object Oriented Programming. They are an abstraction, encapsulation, inheritance, and polymorphism. Basically, Java OOP concepts let us create working methods and variables,
- **C# .Net:** C# is no different. It can work with different types of databases. It can work with the most common databases such as Oracle and Microsoft SQL Server. It also can work with new forms of databases such as MongoDB and MySQL.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=wuluqbxxaD4>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No :78-87)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)

Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / II

Course Name with Code : **Relational Database Management Systems / 19CAB03**

Course Faculty : **Mrs. R.Pavithra**

Unit : **V -SPECIAL PURPOSE DATABASES**

Date of Lecture:12.11.2020

Topic of Lecture: Persistence

Introduction :

- A persistent database stores persistent data in the form of objects, or records that are durable when changing devices and software. Persistent data is stable and recoverable.
- Traditional relational database management systems (RDBMS) store persistent data in the form of records and tables.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Memory
- Software
- Data Model
- Records

Detailed content of the Lecture:

Persistence:

- Persistent data in the field of data processing denotes information that is infrequently accessed and not likely to be modified.
- Static data is information, for example a record, that does not change and may be intended to be permanent. It may have previously been categorized as persistent or dynamic.
- A persistent database stores persistent data in the form of objects, or records that are durable when changing devices and software.
- Persistent data is stable and recoverable. Traditional relational database management systems (RDBMS) store persistent data in the form of records and tables.
- Persistence is "the continuance of an effect after its cause is removed".
- In the context of storing data in a computer system, this means that the data survives after the process with which it was created has ended. In other words, for a data store to be considered persistent, it must write to non-volatile storage.

Need persistence in your data store, then you need to also understand the four main design approaches that a data store can take and how (or if) these designs provide persistence:

- Pure in-memory, no persistence at all, such as memcached or Scalaris
- In-memory with periodic snapshots, such as Oracle Coherence or Redis
- Disk-based with update-in-place writes, such as MySQL ISAM or MongoDB
- Commit log-based, such as all traditional OLTP databases (Oracle, SQL Server, etc.)
- In-memory approaches can achieve blazing speed, but at the cost of being limited to a relatively small data set.
- Most workloads have relatively small "hot" (active) subset of their total data; systems that require the whole data set to fit in memory rather than just the active part are fine for caches but a bad fit for most other applications.
- Because the data is in memory only, it will not survive process termination.
- Therefore these types of data stores are not considered persistent.

Persistence of Objects:

- **Persistence by Classes:** Persistent classes are classes in an application that implement the entities of the business problem Persistent classes have, as the name implies, transient and also persistent instance stored in the database.
- **Persistence by Creation :** Introduce new syntax to create persistent Objects.
- **Persistence by Marking:** Mark an object persistent after it is create.
- **Persistence by Reference :**One or more objects are explicitly declared as (root) persistent objects. All other objects are persistent iff they are referred, directly or indirectly, from a root persistent object. It is easy to make the entire data structure persistent by merely declaring the root of the structure as persistent, but is expensive to follow the chains in detection for a database system.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=ZBPcl4-WCag>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No :36-41)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)

Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : V -SPECIAL PURPOSE DATABASES

Date of Lecture:13.11.2020

Topic of Lecture: Object Relational Databases

Introduction :

- An object–relational database (ORD), or object–relational database management system (ORDBMS), is a database management system (DBMS) similar to a relational database, but with an object-oriented database model: objects, classes and inheritance are directly supported in database schemas and in the query language.

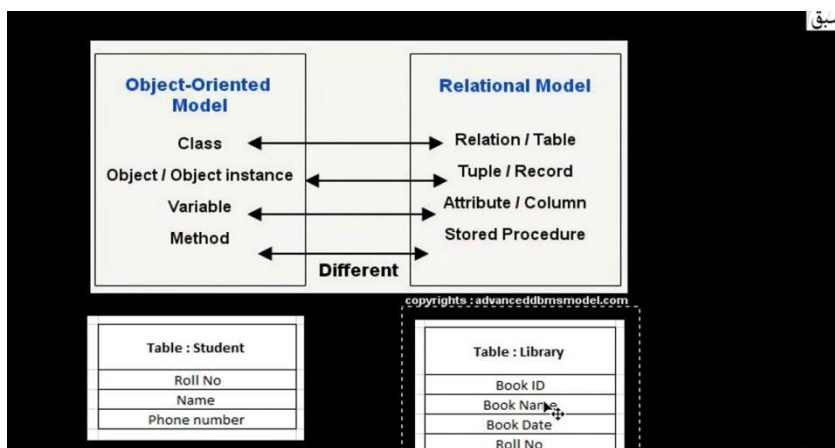
Prerequisite knowledge for Complete understanding and learning of Topic:

- Query Languages, API, Database, Schema

Detailed content of the Lecture:

Object Relational Databases :

- An object–relational database can be said to provide a middle ground between relational databases and object-oriented databases.
- In object–relational databases, the approach is essentially that of relational databases: the data resides in the database and is manipulated collectively with queries in a query language; at the other extreme are OODBMSes in which the database is essentially a persistent object store for software written in an object-oriented programming language, with a programming API for storing and retrieving objects, and little or no specific support for querying.



Advantages of Object Relational model:

- Inheritance. The Object Relational data4 model allows its users to inherit objects, tables etc.
- Complex Data Types. Complex data types can be formed using existing data types.
- Extensibility. The functionality of the system can be extended in Object relational data model.

Disadvantages of Relational Database:

- Cost. The underlying cost involved in a relational database is quite expensive.
- Performance. Always the performance of the relational database depends on the number of tables.

- Physical Storage.
- Complexity.
- Information Loss.
- Structure Limitations.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=Oszv7Ngd6AA>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No :221-227)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : V -SPECIAL PURPOSE DATABASES

Date of Lecture: 16.11.2020

Topic of Lecture: XML

Introduction :

- Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The World Wide Web Consortium's XML 1.0 Specification .

Prerequisite knowledge for Complete understanding and learning of Topic:

- Markup Language.
- Design
- WWW
- HTML

Detailed content of the Lecture:

XML:

- The Extensible Markup Language (XML) is a simple text-based format for representing structured information documents, data, configuration, books, transactions, invoices, and much more.
- It was derived from an older standard format called SGML (ISO 8879), in order to be more suitable for Web use.
- XML schema that defines the contents of an Excel workbook, including XML tags that store all workbook information, such as data and properties, and define the overall structure of the workbook. Custom applications can use this Excel macro-enabled Office XML Format File.

To download as an XML file:

- In an open table or worksheet, click Download > As XML.
- Select the download format: Option. Description. Not compressed. Download the current table or worksheet as an XML file named download.xml.
- Compressed. Download the current table or worksheet as an XML file named download.zip.
- It is possible to create a sample XML document from an XML Schema.
- This can be done by using the "Tools->Create Sample XML Document" Menu ().

- The sample XML Document is created by stepping through the possible options in the schema and creating the appropriate elements and attributes as it goes.
- paramount importance in large projects.

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type='text/xsl'?>
<CATALOG>
  <CD>
    <TITLE>dill diya galla</TITLE>
    <ARTIST>Arijit singh</ARTIST>
    <COUNTRY>India</COUNTRY>
    <COMPANY>tseries</COMPANY>
    <PRICE>10.90</PRICE>
    <YEAR>2018</YEAR>
  </CD>
</CATALOG>
```

Components of XML Document:

- Document Prolog.
- Document Instance.
- Processing Instruction.
- Comments.

Advantages of XML:

- Definition of own Tags.
- Different Views.
- Platform Independence.
- Ease of Information Exchange.
- Standards Creation.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=n-y-YHVZSwk>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No :76-87)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : V -SPECIAL PURPOSE DATABASES

Date of Lecture: 17.11.2020

Topic of Lecture: Temporal Databases

Introduction :

- A temporal database stores data relating to time instances.
- It offers temporal data types and stores information relating to past, present and future time.
- Temporal databases could be uni-temporal, bi-temporal or tri-temporal.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Instance
- Temporal
- Integrity

Detailed content of the Lecture:

Temporal Databases :

- . A temporal database is a collection of time-referenced data.
- In such a database, the time references capture some temporal aspect of the data; put differently, the data are timestamped. Two temporal aspects are prevalent.

More specifically the temporal aspects usually include:

- **Valid time** is the time period during which a fact is true in the real world.
- **Transaction time** is the time at which a fact was recorded in the database.
- **Decision time** is the time at which the decision was made about the fact.

Features of Temporal Database:

- A time period datatype, including the ability to represent time periods with no end (infinity or forever)
- The ability to define valid and transaction time period attributes and bitemporal relations
- System-maintained transaction time
- Temporal primary keys, including non-overlapping period constraints
- Temporal constraints, including non-overlapping uniqueness and referential integrity
- Update and deletion of temporal records with automatic splitting and coalescing of time periods

- Temporal queries at current time, time points in the past or future, or over durations
- Predicates for querying time periods, often based on Allen's interval relations

APPLICATION DOMAINS OF TEMPORAL DATA:

- Financial Applications.
- Reservation Systems.
- Medical Systems.
- Computer Applications .
- Archive Management Systems.
- Health Care.
- Insurance.
- Reservation.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=t4P76BMJXHs>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No :56-67)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03

Course Faculty : Mrs. R.Pavithra

Unit : V -SPECIAL PURPOSE DATABASES

Date of Lecture: 18.11.2020

Topic of Lecture: Mobile Databases

Introduction :

- A Mobile database is a database that can be connected to a mobile computing device over a mobile network (or wireless network). Here the client and the server have wireless connections.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Wireless Communication
- PDA
- Wireless Network
- Memory

Detailed content of the Lecture:

Mobile Databases:

- A mobile database is a database that resides on a mobile device such as a PDA, a smart phone, or a laptop.
- Such devices are often limited in resources such as memory, computing power, and battery power.

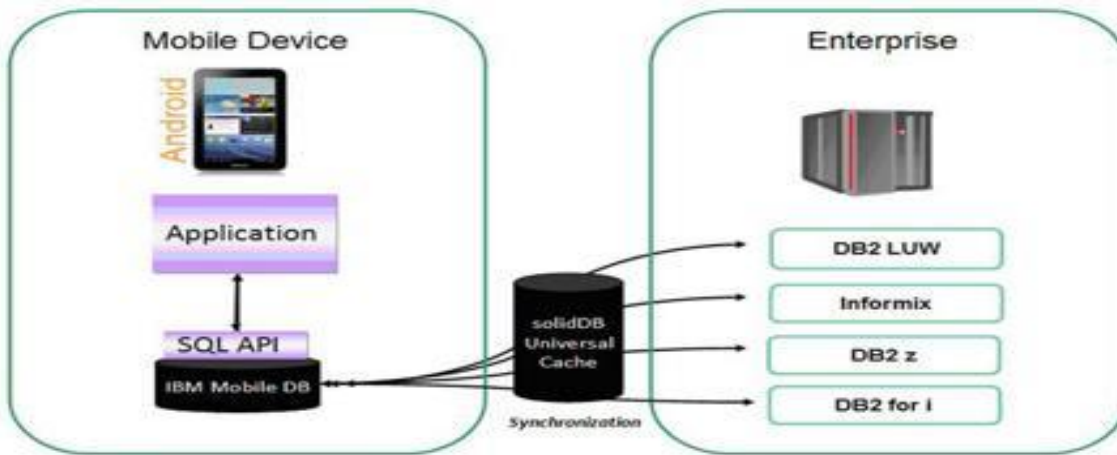
CHARACTERISTICS OF MOBILE COMPUTING:

- **Portability** - The Ability to move a device within a learning environment or to different environments with ease.
- **Social Interactivity** - The ability to share data and collaboration between users.
- **Context Sensitivity** - The ability to gather and respond to real or simulated data unique to a current location, environment, or time.
- **Connectivity** - The ability to be digitally connected for the purpose of communication of data in any environment.
- **Individual** - The ability to use the technology to provide scaffolding on difficult activities and lesson customization for individual learners.
- **Small Size** - Mobile devices are also known as handhelds, palmtops and smart phones due

to their roughly phone-like dimensions.

- A typical mobile device will fit in the average adult's hand or pocket.
- **Wireless Communication** - Mobile devices are typically capable of communication with other similar devices, with stationary computers and systems, with networks and portable phones.
- Base mobile devices are capable of accessing the Internet through Bluetooth or Wi-Fi networks, and many models are equipped to access cell phone and wireless data networks as well.

Mobile Database System Architecture:



- A mobile database is a database that can be connected to by a mobile computing device over a mobile network.
- A database that is portable and physically separate from the corporate database server.

Components:

- Corporate Database Server.
- Mobile Database.
- Mobile Database Platform.
- Communication Links.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=BWvncI0K7WE>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No :197-201)

Course Faculty

Verified by HOD



MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu



LECTURE HANDOUTS

MCA

I / I

Course Name with Code : Relational Database Management Systems / 19CAB03
Course Faculty : Mrs. R.Pavithra
Unit : V -SPECIAL PURPOSE DATABASES

Date of Lecture: 19.12.2020

Topic of Lecture: Spatial Databases

Introduction :

- Spatial data represents information about the physical location and shape of geometric objects.
- These objects can be point locations or more complex objects

Prerequisite knowledge for Complete understanding and learning of Topic:

- Records, Indexing, Query Language , Geometric

Detailed content of the Lecture:

Spatial Databases:

- A spatial database system is a database system.
- It offers spatial data types (SDTs) in its data model and query language.
- It supports spatial data types in its implementation, providing at least spatial indexing and efficient algorithms for spatial join.

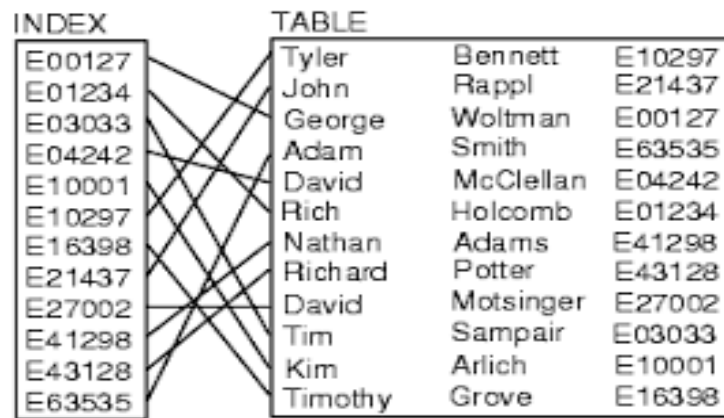
Features of Spatial Databases:

- **Spatial Measurements:** Computes line length, polygon area, the distance between geometries, etc.
- **Spatial Functions:** Modify existing features to create new ones, for example by providing a buffer around them, intersecting features, etc.
- **Spatial Predicates:** Allows true/false queries about spatial relationships between geometries. Examples include "do two polygons overlap" or "is there a residence located within a mile of the area we are planning to build the landfill".
- **Geometry Constructors:** Creates new geometries, usually by specifying the vertices (points or nodes) which define the shape.
- **Observer Functions:** Queries which return specific information about a feature such as the location of the center of a circle.

Spatial Indexing:

- A spatial index is a type of extended index that allows you to index a spatial column. A spatial column is a table column that contains data of a spatial data type, such as geometry or

geography.



Spatial Database Queries:

- Range Query.
- Nearest Neighbour Query.
- Spatial Joins or Overlays.

Techniques for Spatial Database Queries:

- R-Tree.
- Quad tree.

Advantages:

- Centralized data store.
- Continuous data sets.
- Improved data Integrity.
- Provide Backup.
- Edit control and archiving.

Video Content / Details of website for further learning (if any):

<https://www.youtube.com/watch?v=qH2-xwUz3tU>

Important Books/Journals for further learning including the page nos.:

Database System Concepts –Abraham Silberschatz, Henry F.Korth and S.Sundarshan , McGraw Hil, Sixth Edition, 2010 (Page No :204-208)

Course Faculty

Verified by HOD