

Introduction to Object Oriented Database

Compiled By: Ram Datta Bhatta

Concept of Object

- ✓ An entity in the real world
- ✓ Object has attribute or what the object has (state)
- ✓ Object has methods or what the object can do (behavior)
- ✓ An object has five aspects: identifier, name, lifetime, structure, and creation.
 - The **object identifier** is a unique system-wide identifier (or Object_id). Every object must have an object identifier (**object identity** is a fundamental concept that refers to the unique identification of individual objects in the database)
 - The **lifetime** of an object specifies whether it is a *persistent object* (that is, a database object) or *transient object*
 - The **structure** of an object specifies how the object is constructed by using the type constructors. The structure specifies whether an object is *atomic* or not. An **atomic object** refers to a single object that follows a user-defined type, such as Employee or Department. If an object is not atomic, then it will be.
 - Object **creation** refers to the manner in which an object can be created. This is typically accomplished via an operation *new*

Object-Oriented Programming Concepts

Object-oriented databases closely relate to object-oriented programming concepts. The four main ideas of object-oriented programming are:

- ✓ **Class and Object**
- ✓ **Polymorphism**
- ✓ **Inheritance**
- ✓ **Encapsulation**
- ✓ **Abstraction**

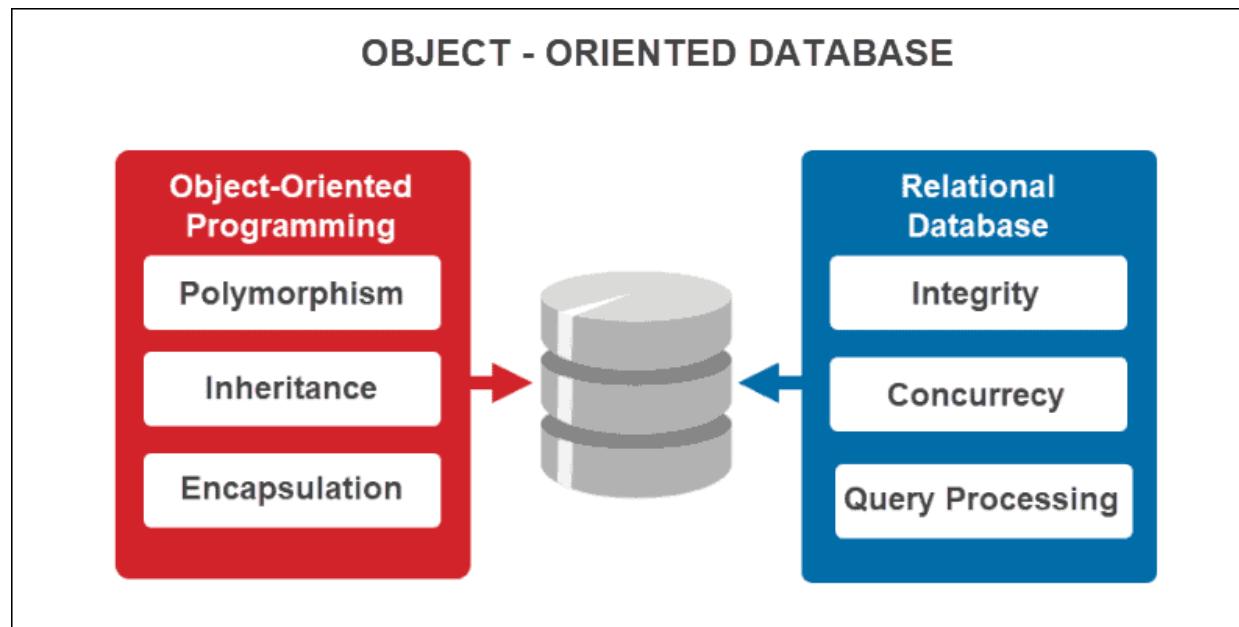
Concept of Object Database

- ✓ An **object database** (also **object-oriented database management system** - OODBMS) is a database management system in which information is represented in the form of objects as used in object-oriented programming.
- ✓ Object-oriented database management systems (OODBMSs) combine database capabilities with [object-oriented programming](#) language capabilities.
- ✓ OODBMSs allow object-oriented programmers to develop the product, store them as objects, and replicate or modify existing objects to make new objects within the OODBMS.
- ✓ Because the database is integrated with the programming language, the programmer can maintain consistency within one environment, in that both OODBMS and the programming language will use the same model of representation. Relational DBMS projects, by way of contrast, maintain a clearer division between the database model and the application.
- ✓ Object databases store objects rather than data such as integers, strings or real numbers. Objects are used in object oriented languages such as Smalltalk, C++, Java, and others.

Objects basically consist of.....

- ✓ Attributes - Attributes are data which defines the characteristics of an object. This data may be simple such as integers, strings, and real numbers or it may be a reference to a complex object.
- ✓ Methods - Methods define the behavior of an object and are what was formally called procedures or functions.
- ✓ **Object-oriented databases** add the [database](#) functionality to object programming languages, creating more manageable code bases.
- ✓ They represent data in the form of object and classes.
- ✓ OODB follow the fundamental principles of OOP.
- ✓ **OOP + Relational Database= ORDB model**

Object Oriented Database



- ✓ In Object oriented database, the data is stored in the form of objects. The data is stored along with its actions that processes or reads the existing data.
- ✓ OO databases try to maintain a direct correspondence between real world and database objects so that objects don't lose their integrity and identify and can easily be identified and operated upon.
- ✓ OODBMS is pure OO while ORDBMS extends RDBMS with OO

Stonebraker's Application Matrix

	No Query	Query
Complex Data	OODBMS	ORDBMS
Simple Data	File System	RDBMS

Some examples of OODBMS are Versant Object Database, Objectivity/DB, ObjectStore, Caché and ZODB.

Object Identifier (OID)

- ✓ An object has two components: state (value) and behavior (operations). When a program terminates then all the objects that is in the primary memory are lost. These kind of objects called **transient object**. They do not live forever. However, OO allows us to store objects permanently in the database, these objects are called **persistent objects**. Because they persist the beyond the life of program. OIDs is the mechanism to refer to persistent objects.
- ✓ An ODMS provides a unique identity to each independent object stored in the database. This unique identity is typically implemented via a unique, **system-generated object identifier (OID)**. The value of an OID is not visible to the external user, but is used internally by the system to identify each object uniquely and to create and manage inter-object reference.

Main advantages of OODBMS

- ✓ Complex data and a wider variety of data types,
- ✓ Easy to save and retrieve data quickly.
- ✓ Seamless integration with object-oriented programming languages.
- ✓ Easier to model the advanced real world problems.
- ✓ Extensible with custom data types.

Main limitations of OODBMS

- ✓ Not as widely adopted as relational databases.
- ✓ No universal data model. Lacks theoretical foundations and standards.
- ✓ Does not support views.
- ✓ High complexity causes performance issues.
- ✓ An adequate security mechanism and access rights to objects do not exist

RDBMS vs OODBMS

<https://www.geeksforgeeks.org/difference-between-rdbms-and-oodbms/>

BASIS	RDBMS	OODBMS
Long Form	Stands for Relational Database Management System.	Stands for Object Oriented Database Management System.
Way of storing data	Stores data in Entities, defined as tables hold specific information.	Stores data as Objects.
Data Complexity	Handles comparatively simpler data.	Handles larger and complex data than RDBMS.
Grouping	Entity type refers to the collection of entity that share a common definition.	Class describes a group of objects that have common relationships, behaviors, and also have similar properties.
Data Handling	RDBMS stores only data.	Stores data as well as methods to use it.
Main Objective	Data Independence from application program.	Data Encapsulation.
Key	A Primary key distinctively identifies an object in a table..	An object identifier (OID) is an unambiguous, long-term name for any type of object or entity.

Object Oriented Extension to SQL (Object Relational Database)

- ✓ Object oriented technology is influencing many areas of software development including database systems.
- ✓ Extensions to SQL in the 1999 and 2003 standards include support for some object-oriented concepts. The data model used in these standards is called **the object-relational data model**.
- ✓ This model includes all the traditional ideas found in the relational data model and includes some ideas from the object data model.
- ✓ The object oriented additions to SQL include features that enable the creation of classes, class hierarchies and objects.
- ✓ The object-relational data model is not a pure object data model because the top-level items in the database are restricted to be tables. Tables can include object values as attributes and typed tables include only objects.

Object Relational Database (Oracle) examples

Question : use of Type (Class and object features in relational database)

- ✓ create type Employee as object (empID number, empName varchar(20)) not final;
 - ✓ create table Employee_t of Employee
 - ✓ insert into employee_t values(9,'hari')
 - ✓ insert into employee_t values (new Employee(8,'hariddd'))
 - ✓ select * from employee_t
- [use oracle express edition to run the queries]

Inheritance in SQL

- ✓ In SQL, inheritance refers to a database design concept that allows one table to inherit properties and attributes from another table. This is a feature found in some object-oriented database management systems (OODBMS) or in databases that support object-relational features.
- ✓ Inheritance in SQL is similar to inheritance in object-oriented programming, where a subclass inherits attributes and behaviors from a superclass. Similarly, in SQL database inheritance, a child table (subclass) inherits attributes and data from a parent table (superclass).

Question : Implementation of Inheritances in relational database

- ✓ create type Person as object (ssn number) not final;
- ✓ create table Person_t of person;
- ✓ create type student under person (rollno number) not final;
- ✓ create table student_t of student;
- ✓ create type Employee under person (salary number, post varchar(10)) not final;
- ✓ create table employee_t of Employee;

Then describe tables, insert values etc

Question : Use of member function / method in relational database

✓ create or replace type Box as object
(l integer,
w integer,
h integer,
member function Volume return integer);

✓ create table box_table of Box;
✓ insert into box_table values(10,20,30);
✓ select bb.volume() from box_table bb;

✓ create or replace type body Box as
member function volume return integer is
begin
return l*w*h;
end;
end;

ORDBMS vs OODBMS

ORDBMS	OODBMS
It is capable of handling massive volumes of data.	It can handle more extensive and more complicated data sets.
Tables (with rows and columns) are used to store data.	Objects are used to store data.
Data is kept in tables, which are made up of rows and columns.	Objects contain data. Objects hold information.
A primary key identifies an object in a table uniquely.	Object Id, OID represents an object uniquely in group of objects.
Extension to SQL (RDBMS). Example: Oracle	Extension to OOP language techniques into the field of persistent data management. Examples: Versant Object Database, Objectivity/DB, ObjectStore

Thank you !!