

Definition and Overview of ODBMS

Difficulty Level : Medium Last Updated : 06 Mar, 2023

[Read](#) [Discuss](#)

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. These classes and objects together make an object-oriented data model.

Components of Object-Oriented Data Model:

The OODBMS is based on three major components, namely: Object structure, Object classes, and Object identity. These are explained below.

1. Object Structure:

The structure of an object refers to the properties that an object is made up of. These properties of an object are referred to as an attribute. Thus, an object is a real-world entity with certain attributes that makes up the object structure. Also, an object encapsulates the data code into a single unit which in turn provides data abstraction by hiding the implementation details from the user.

The object structure is further composed of three types of components: Messages, Methods, and Variables. These are explained below.

1. Messages –

A message provides an interface or acts as a communication medium between an object and the outside world. A message can be of two types:

- **Read-only message:** If the invoked method does not change the value of a variable, then the invoking message is said to be a read-only message.
- **Update message:** If the invoked method changes the value of a variable, then the invoking message is said to be an update message.

2. Methods –

When a message is passed then the body of code that is executed is known as a method. Whenever a method is executed, it returns a value as output. A method can be of two types:

- **Read-only method:** When the value of a variable is not affected by a method, then it is known as the read-only method.
- **Update-method:** When the value of a variable change by a method, then it is known as an update method.

3. Variables –

It stores the data of an object. The data stored in the variables makes the object distinguishable from

one another.

2. Object Classes:

An object which is a real-world entity is an instance of a class. Hence first we need to define a class and then the objects are made which differ in the values they store but share the same class definition. The objects in turn correspond to various messages and variables stored in them.

Example –

```
class CLERK

{ //variables
    char name;
    string address;
    int id;
    int salary;

    //Messages
    char get_name();
    string get_address();
    int annual_salary();
};
```

In the above example, we can see, CLERK is a class that holds the object variables and messages.

An OODBMS also supports inheritance in an extensive manner as in a database there may be many classes with similar methods, variables and messages. Thus, the concept of the class hierarchy is maintained to depict the similarities among various classes.

The concept of encapsulation that is the data or information hiding is also supported by an object-oriented data model. And this data model also provides the facility of abstract data types apart from the built-in data types like char, int, float. ADT's are the user-defined data types that hold the values within them and can also have methods attached to them.

Thus, OODBMS provides numerous facilities to its users, both built-in and user-defined. It incorporates the properties of an object-oriented data model with a database management system, and supports the concept of programming paradigms like classes and objects along with the support for other concepts like encapsulation, inheritance, and the user-defined ADT's (abstract data types).

ODBMS stands for Object-Oriented Database Management System, which is a type of database management system that is designed to store and manage object-oriented data. Object-oriented data is

data that is represented using objects, which encapsulate data and behavior into a single entity.

An ODBMS stores and manages data as objects, and provides mechanisms for querying, manipulating, and retrieving the data. In an ODBMS, the data is typically stored in the form of classes and objects, which can be related to each other using inheritance and association relationships.

In an ODBMS, the data is managed using an object-oriented programming language or a specialized query language designed for object-oriented databases. Some of the popular object-oriented database languages include Smalltalk, Java, and C++. Some ODBMS also support standard SQL for querying the data.

ODBMS have several advantages over traditional relational databases. One of the main advantages is that they provide a natural way to represent complex data structures and relationships. Since the data is represented using objects, it can be easier to model real-world entities in the database. Additionally, ODBMS can provide better performance and scalability for applications that require a large number of small, complex transactions.

However, there are also some disadvantages to using an ODBMS. One of the main disadvantages is that they can be more complex and harder to use than traditional relational databases. Additionally, ODBMS may not be as widely used and supported as traditional relational databases, which can make it harder to find expertise and support. Finally, some applications may not require the advanced features and performance provided by an ODBMS, and may be better suited for a simpler database solution.