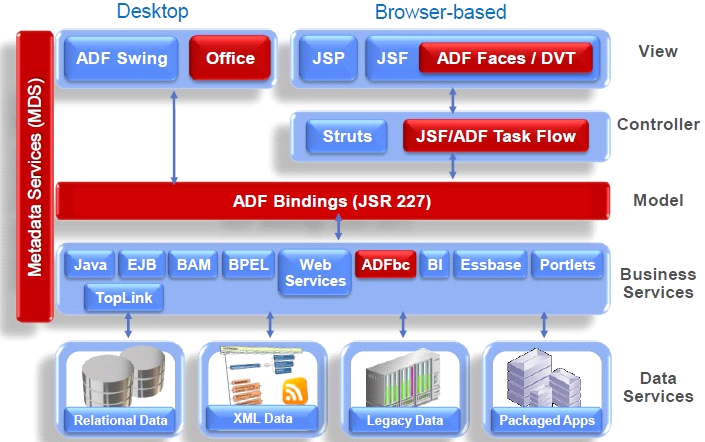
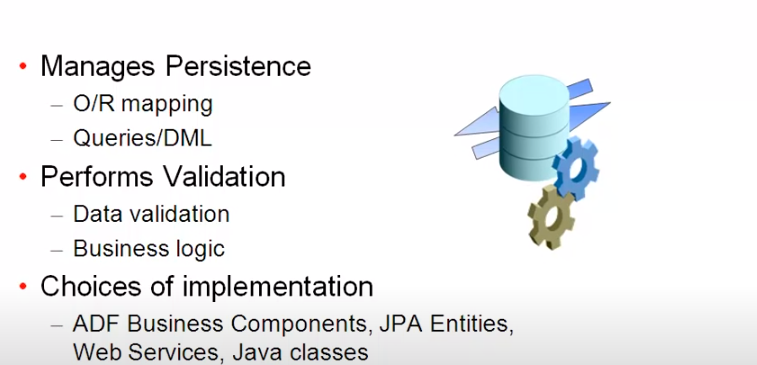
**Introduction to Oracle ADF Business Components**



The Business Service Layer



Introduction to Oracle ADF Business Components

Overview of ADF Business Components

- ADF Business Components is a framework for building Java EE based business services that interact with a database

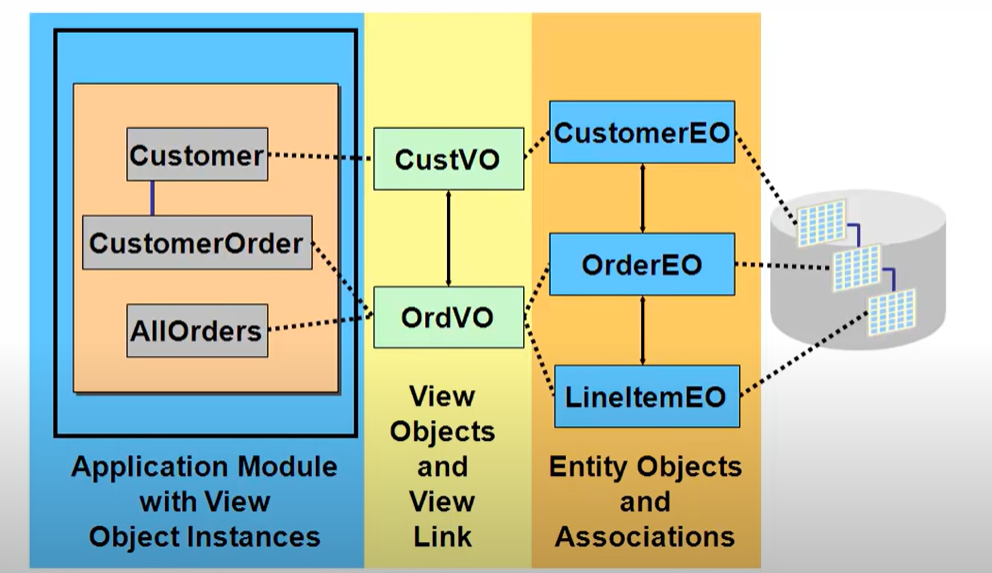
- It simplifies data access and writing business logic on top of data

- Development is mostly declarative through metadata rather than code

- Provides separation of data views from business logic and validation

- Implements best practices and design patterns for Java EE applications

**Artifacts of ADF Business Components**



• ADF business components are divided into three layers: entity objects, view objects, and application model.

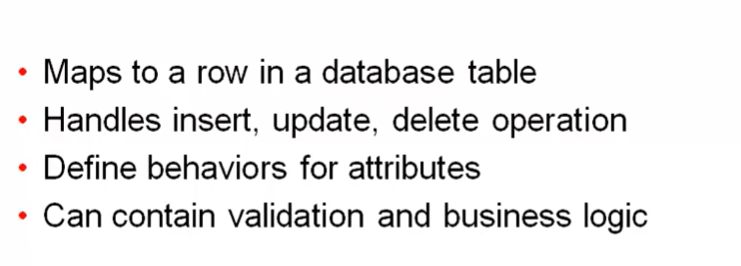
• Entity objects represent database tables and allow operations like insert, update, and delete.

• View objects represent data sets that can be updated or non-updatable.

• The application model exposes the interface for user interface designers and handles database connection in transaction management.

• The data model is constructed from view objects, view links, and service methods at the application module level.

**Entity Objects**

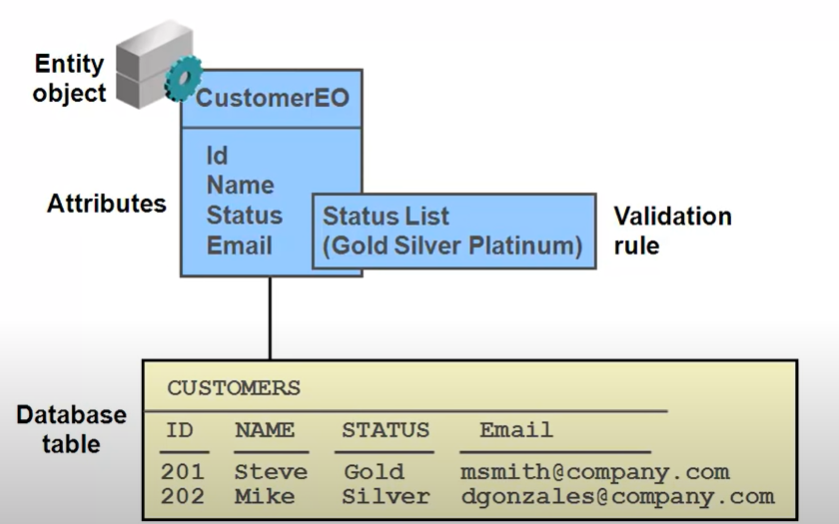


- Represent tables in the database and map to database tables

- Consist of attributes that map to columns

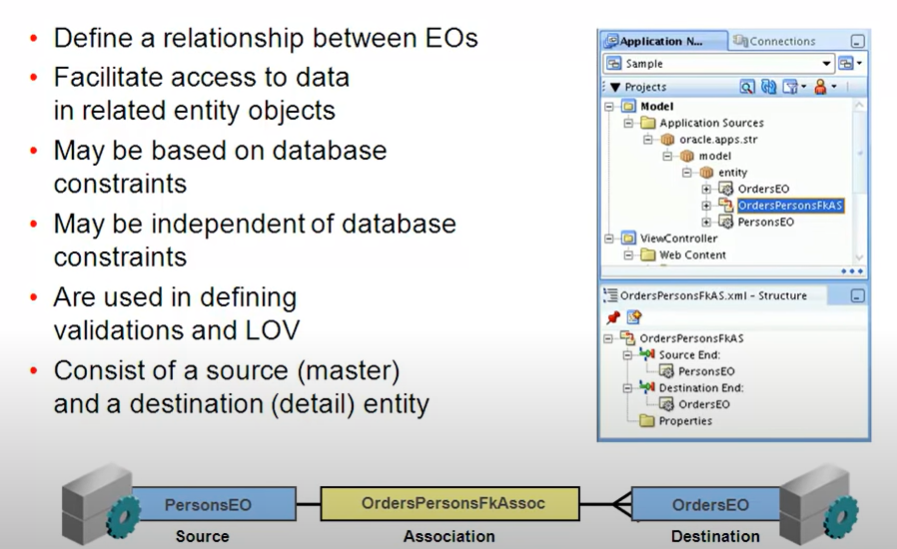
- Handle insert, update, delete operations against the table

- Define behaviors and validation for attributes



**Association**

Association represents relationships between entity objects.



Associations in Database Management

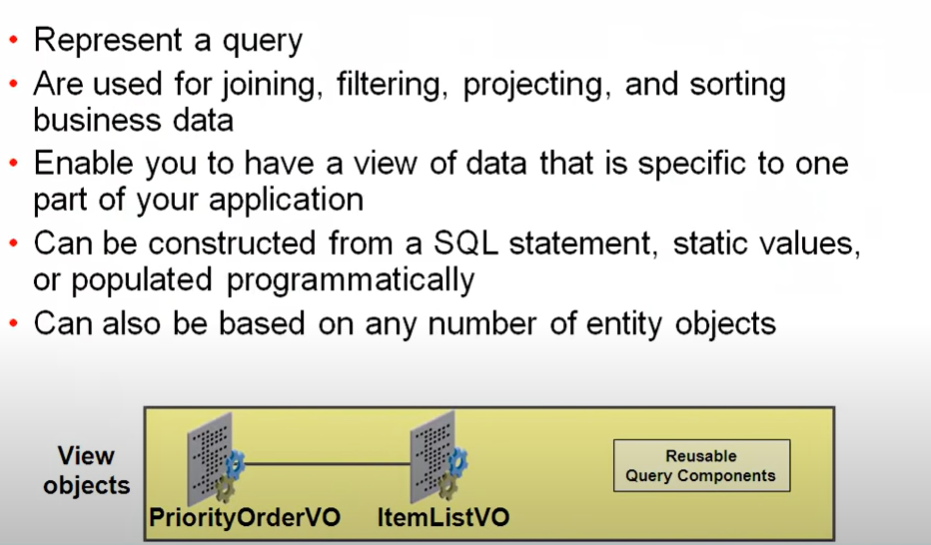
• Represent relationships between entity objects.

• Map to database relationships.

• Can be independent of database relationships.

• Used to access data from one entity to another.

**View Objects**

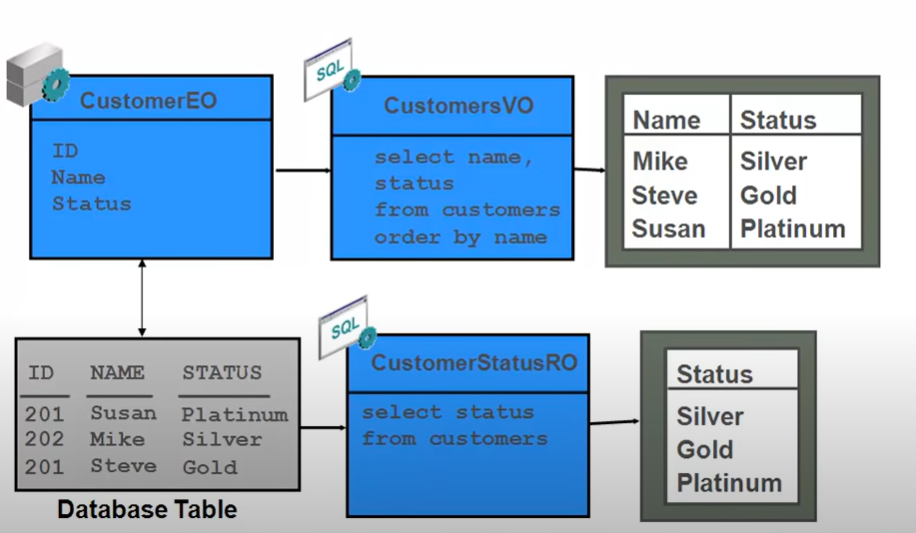


- Represent queries against data

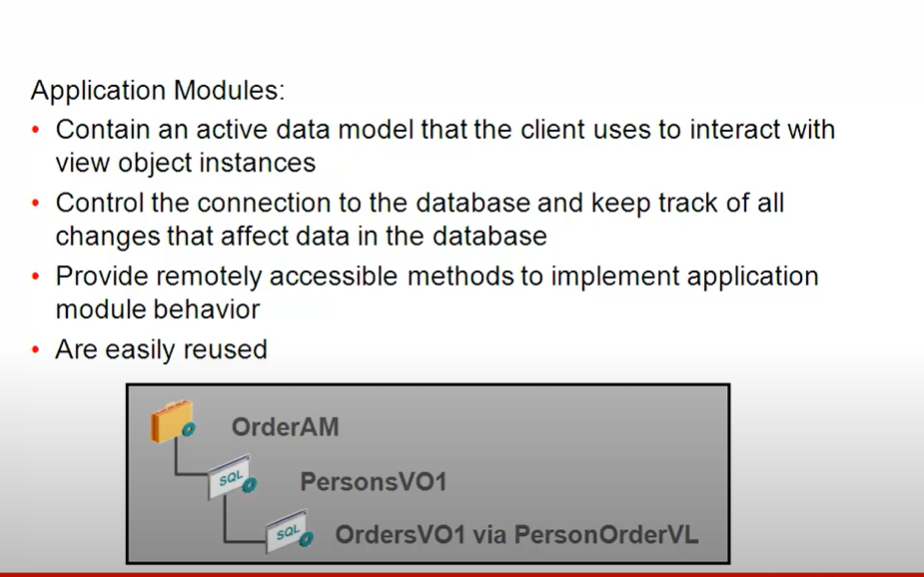
- Can be based on single or multiple entity objects

- Can be updatable or non-updatable

- Define specific sets of data through filtering, joining, etc.

****

**Application Model**

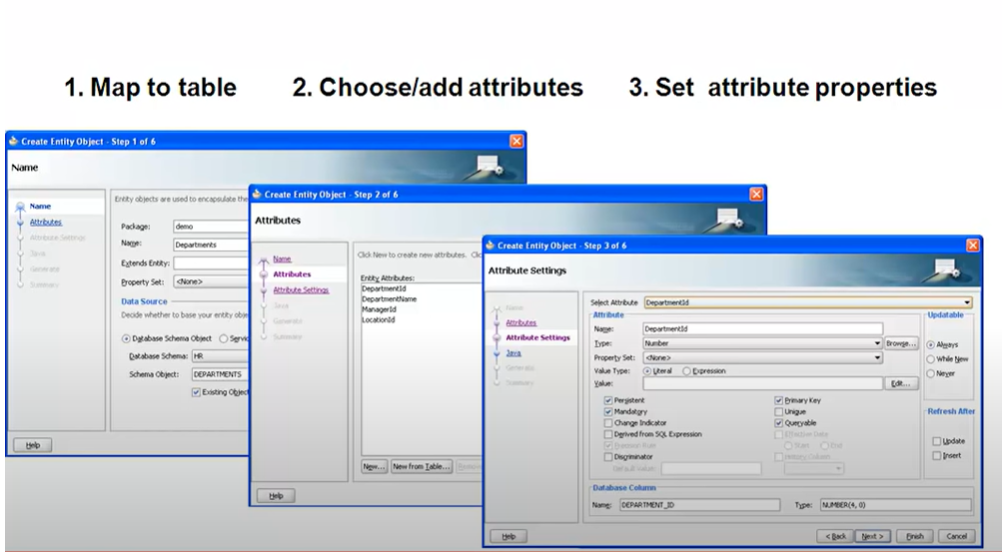


- Exposes data model for developers to use

- Controls database connection and transaction management

- Exposes view objects, view links, and service methods

**Creating Entity Objects**

****