Introduction to erp:

ERP : Enterprise Resource Planning

Software application to maintain enterprise resources

Enterprise – A bigger organization

Resources - People , Money, Machinery, Documents etc.

ERP Vendors

Vendor	Product Name	Cloud/OnPremise
Oracle	E-Business Suite	OnPremise
Oracle	Cloud/Fusion ERP	Cloud
Workday	Workday	Cloud
Microsoft	MS Dynamics	OnPremise
Infor	Infor	Cloud
Coupa	Coupa	Cloud
Ariba	Ariba	Cloud
SAP	S/4 Hana	Cloud/OnPremise
SAP	SAP Business One	OnPremise

Types of ERP systems :

1. On-Premise ERP

- Installed on the company's own servers. The organization manages and maintains the software.

2. Cloud ERP

- Hosted on the internet (cloud) and accessed via a web browser. The provider manages everything, making it easier to update and scale.

3. Hybrid ERP

- A mix of on-premise and cloud solutions. Some parts are kept in-house, while others are managed in the cloud.

Common ERP Modules (EBS)

1. **Financial Management (FM)**
    a. Oracle General Ledger (GL), Oracle Accounts Payable (AP)
2. **Supply Chain Management (SCM)**
    a. Oracle Purchasing (PO), Oracle Order Management (OM)
3. **Human Resources Management (HRM)**
    a. Oracle Human Resources (HR), Oracle Payroll
4. **Customer Relationship Management (CRM)**
    a. Oracle Sales (SFA), Oracle Marketing
5. **Manufacturing (MFG)**
    a. Oracle Manufacturing
6. **Project Management (PM)**
    a. Oracle Projects
7. **Sales and Marketing (S&M)**
    a. Oracle Order Management (OM)
8. **Inventory Management (IM)**
    a. Oracle Inventory (INV)

Implementation methodology of oracle ERP (SDLC) :

- **Waterfall**: Best for projects with clearly defined requirements, ensuring a structured implementation of Oracle EBS.
- **Hybrid Agile**: Offers flexibility and responsiveness, making it suitable for projects where user needs may evolve during implementation.

Types of Oracle E-Business Projects

1. **Implementation**
    a. Establishing Oracle E-Business Suite from scratch, configuring modules to meet business needs.
2. **Upgrade**

a. Moving from an older version of Oracle EBS to a newer version, enhancing features and functionalities.

3. **Rollout**
    a. Expanding Oracle EBS to new departments or geographical locations within the same organization. E.g. amazon coming from usa to india.
4. **Support**
    a. Ongoing assistance and maintenance for existing Oracle EBS systems, including troubleshooting and user support.

Roles in E-Business Projects

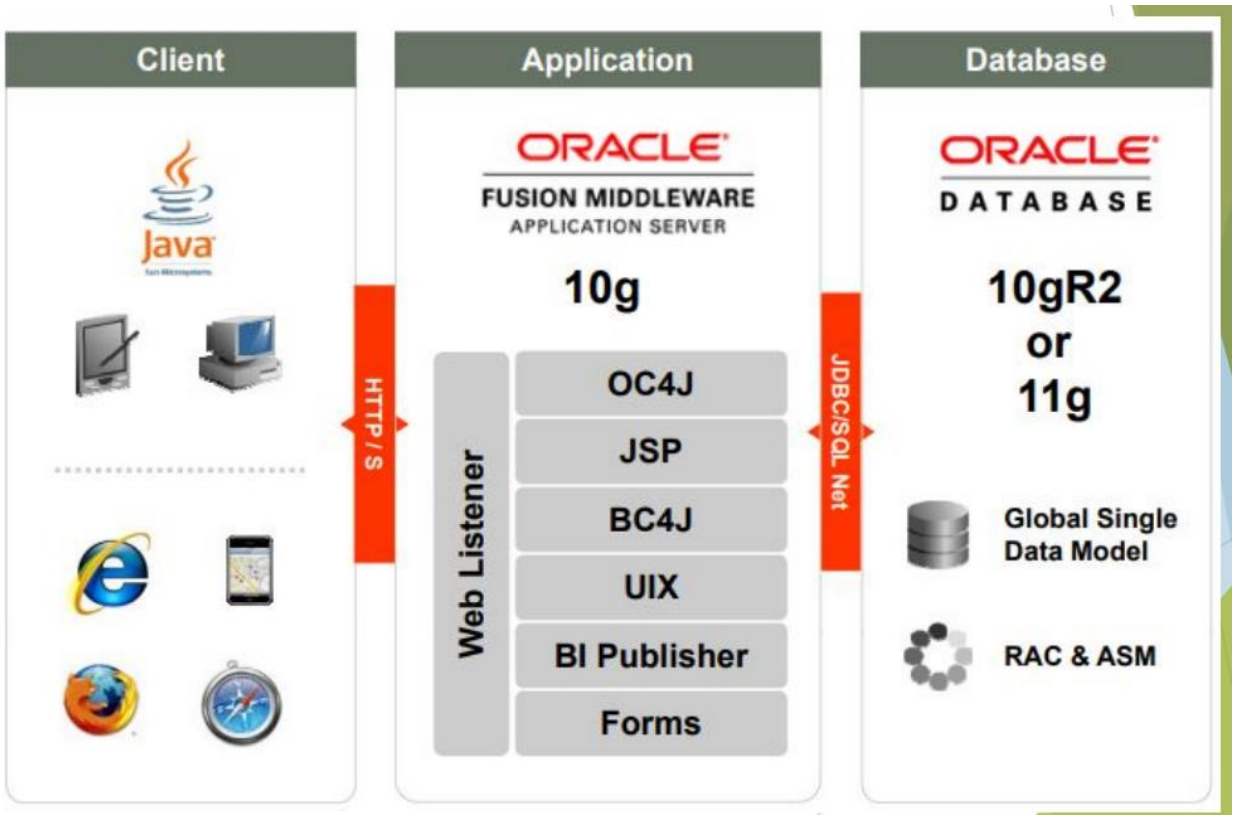
1. **Technical**
    a. Focuses on the technical aspects of Oracle EBS, including software installation, configuration, and integration with other systems.
2. **Functional**
    a. Specializes in business processes and requirements, ensuring that the Oracle EBS modules are configured to meet user needs and business objectives.
3. **Techno-Functional**
    a. Combines technical and functional expertise, bridging the gap between IT and business teams. Responsible for both system configuration and technical support.
4. **DBA (Database Administrator)**
    a. Manages the Oracle database, ensuring data integrity, performance, and security. Responsible for backups, recovery, and database tuning.

EBS tools :

Component Type	Tool	Programming Language	Purpose
Reports (RDF)	Oracle Report Builder	SQL/PLSQL	Create business reports (e.g., sales reports)
Reports (XML)	Oracle Template Builder for Word	SQL/PLSQL/XML	Create business reports (e.g., invoices)
Interfaces	SQL Developer / TOAD / SOAP UI	SQL/PLSQL / SQLLoader	Connect with other systems (e.g., share data with CRM, ERP, or accounting software)
Conversions	SQL Developer / TOAD	SQL/PLSQL	Upload data to the system (e.g., adding customer info)
Extensions	Oracle Form Developer / JDeveloper	SQL/PLSQL/Java	Add new features to the software (e.g., new fields in forms)
Workflow	Oracle Workflow Builder		Manage approval processes (e.g., getting manager approval for requests)
Forms	Oracle Form Builder	SQL/PLSQL	Design and extend user forms (e.g., create a form for entering orders)
OAF	JDeveloper	SQL/PLSQL/Java	Design user interface pages (e.g., dashboards for users)

E-Business Architecture :

- **OC4J:** A server that runs Java applications in Oracle EBS.
- **JSP:** A way to create dynamic web pages that display data.
- **BC4J:** A framework for handling business logic and database access in Java.
- **UIX:** A tool for building user interfaces using XML, making web pages interactive.



Application Tier Folder Structure :

1. **$APPL_TOP**
    a. The main directory for the application files.
    b. Contains all application-related subdirectories.

/u01/app/oracle/product/12.2.0/apps/apps_st/appl

Subdirectories:

- **Admin Directory**: Contains administrative scripts and configuration files.
    - **Path**: `/u01/app/oracle/product/12.2.0/apps/apps_st/appl/admin`
- **Binary Directory**: Contains executable files and scripts.
    - **Path**: `/u01/app/oracle/product/12.2.0/apps/apps_st/appl/bin`
- **Clone Directory**: Used for cloning the EBS instance.
    - **Path**: `/u01/app/oracle/product/12.2.0/apps/apps_st/appl/clone`
- **Forms Directory**: Contains Oracle Forms files (`*.fmb`, `*.fmx`).
    - **Path**: `/u01/app/oracle/product/12.2.0/apps/apps_st/appl/forms`
- **Reports Directory**: Contains Oracle Reports files (`*.rdf`, `*.xml`).
    - **Path**: `/u01/app/oracle/product/12.2.0/apps/apps_st/appl/reports`

- **Technology Version Directory**: Contains technology-specific files.
    - **Path**: `/u01/app/oracle/product/12.2.0/apps/apps_st/appl/tech_version`
- **Logs Directory**: Contains log files for application processes.
    - **Path**: `/u01/app/oracle/product/12.2.0/apps/apps_st/appl/logs`
- **HTML Directory**: Contains static web content (HTML, CSS, JavaScript).
    - **Path**: `/u01/app/oracle/product/12.2.0/apps/apps_st/appl/htdocs`

E-Business Architecture (Database Tier)

1. **Product Schemas**:
    a. Individual data storage for specific applications.
    b. Example: The GL (General Ledger) schema includes all the tables for managing financial data.
2. **APPS Schema**:
    a. Centralized code and logic for all applications in EBS.
    b. Includes things like PL/SQL scripts, views, and triggers that help the applications function.