



srivenkateshwaraa
COLLEGE OF ENGINEERING & TECHNOLOGY
 (Approved by AICTE, New Delhi & Affiliated to Pondicherry University, Puducherry.)
 13-A, Pandy-Villupuram Main Road, Ariyur, Puducherry-605 102.
 Phone: 0413-2644426, Fax: 2644424 / Website: www.svcetpandy.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Subject Code: **CS T61**

Subject Name: **ENTERPRISE SOLUTIONS**

Prepared By :

Dr. N. Balaji, Professor & HOD / CSE

Verified by:

Approved by:

UNIT I

Introduction : ERP - Definition - Concept - Fundamentals - Need for ERP - Advantages of ERP - Implementation of ERP - Key issues and Characteristics of ERP Typical architecture components of ERP - ERP system Architecture. ERP and related technologies: Business Process RE-engineering - Management Information System- Decision Support System - Executive Support System - On-Line Analytical Processing, Supply Chain Management, Customer Relationship Management.

2 MARKS

1. Define ERP?

Enterprise Resource Planning (ERP) is business management software. It typically a suite of integrated applications that a company can use to collect, store, manage and interpret data from many business activities, including product planning, cost , Manufacturing or service delivery, Marketing and sales, Inventory.

2. What are the fundamental components of ERP?

The following are the fundamental components of the ERP.

- Financial Management
- Business Intelligence
- Supply Chain Management
- Human Resource Management
- Manufacturing Operations
- Integration

3. What are the elements of ERP?

- production planning
- Integrated logistics
- Human resources
- Accounting and financials
- Sales , distribution(order entry)

4. Why companies are forced to have ERP suites?

The forces driving ERP are;

- The need to create a framework that will, improve customer order processing.
- The need to consolidate and unify business functions such as manufacturing finance distributions/logistics, & human resources.
- The need to integrate a broad range of desperate technologies, along with the processes they support.
- The need to create a new foundations an which next-generation applications can be developed.

5. In what way ERP is used in real world?

- Microsoft in the s/w industry
- Owens-Corning in the building supplies industry.
- Colgate-Palmolive in the consumer products industry.

6. What are the implementation methodology phases of accelerated ERP approach?

The Project Preparation Phase

- The Blueprint Phase
- The Pilot Phase
- The Assessment Phase
- The Final Phase

7. What are the capabilities of effective co-ordination management?

- Strategic Thinking
- Process Reengineering
- Managing Implementation Complexity
- Transition Management

8. What are the elements required in ERP to achieve flexibility?

Four crucial elements are required to achieve flexibility:

- 1.) Components, not modules:
- 2.) Incremental migration, rather than massive reengineering

3.) Dynamic, rather than static, configuration of ERP systems.

4.) Management of multiple strategic sourcing and partnership relationships.

9. What are the advantages of ERP?

- Sales order processing lead time was reduced from 1 hour to 10 minutes.
- Purchase order lead time was reduced from 1 to 4 hours to less than 5 minutes.
- Production scheduling run time was reduced from 18 hours to 30 minutes
- Ninety-eight percent of orders are now delivered on time.

10. What is BPR (Business Process Reengineering)?

Business process re-engineering: Business process re-engineering (BPR) is the analysis and redesign of workflows within and between enterprises in order to optimize end-to-end processes and automate non-value-added tasks. Business Process Reengineering (BPR) is the practice of rethinking and redesigning the way work is done to better support an organization's mission and reduce costs. Reengineering starts with a high-level assessment of the organization's mission, strategic goals, and customer needs.

11. Define Management Information System.

A management information system (MIS) is a computerized database of financial information organized and programmed in such a way that it produces regular reports on operations for every level of management in a company.

12. Define Decision Support System.

Decision support system (DSS) is a computer-based information system that supports business or organizational decision-making activities. DSSs serve the management, operations, and planning levels of an organization (usually mid and higher management) and help people make decisions about problems that may be rapidly changing and not easily specified in advance—i.e. Unstructured and Semi-Structured decision problems. Decision support systems can be either fully computerized, human-powered or a combination of both.

13. Define Executive Support System?

An executive information system (EIS), also known as an executive support system (ESS), is a type of management information system that facilitates and supports senior executive information and decision-making needs. It provides easy access to internal and external information relevant to organizational goals. It is commonly considered a specialized form of decision support system (DSS). EIS emphasizes graphical displays and easy-to-use user interfaces.

14. What is On-Line Analytical Processing?

OLAP (Online Analytical Processing) is the technology behind many Business Intelligence (BI) applications. OLAP is a powerful technology for data discovery, including capabilities for limitless report viewing, complex analytical calculations, and predictive “what if” scenario (budget, forecast) planning.

15. Define Supply Chain Management

Supply chain management (SCM) is the management of the flow of goods and services.

It includes the movement and storage of raw materials, work-in-process inventory, and finished goods from point of origin to point of consumption.

16. Define Customer Relationship Management

Customer relationship management (CRM) is an approach to managing a company's interaction with current and future customers. It often involves using technology to organize, automate, and synchronize sales, marketing, customer service, and technical support.

ELEVEN MARK QUESTIONS:

1. Explain in detail about ERP and its Functional Components.

DEFINITION OF ERP: Enterprise Resource Planning (ERP) is business management software. It typically a suite of integrated applications that a company can use to collect, store, manage and interpret data from many business activities, including product planning, cost , Manufacturing or service delivery, Marketing and sales, Inventory .

An important goal of ERP is to facilitate the flow of information so business decisions can be data-driven. ERP software suites are built to collect and organize data from various levels of an organization to provide management with insight into key performance indicators (KPIs) in real time.

ERP is the backbone of e-business. ERP integrates application suites. ERP is not a single system, but a framework that includes administrative apps (finance, accounting), human resource apps (payroll, benefits), and manufacturing resource planning (MRP) apps (procurement, production planning). ERP unites major business processes – order processing, general ledger, payroll, and production within a single family of software modules.

Fundamental Components of ERP: The following are the fundamental components of the ERP.

- Financial Management
- Business Intelligence
- Supply Chain Management

- Human Resource Management
- Manufacturing Operations
- Integration

Financial Management: At the core of ERP are the financial modules, including general ledger, accounts receivable, accounts payable, billing and fixed asset management. If your organization is considering the move to an ERP system to support expansion into global markets, make sure that multiple currencies and languages are supported, as well as regulatory compliance in the U.S. and in foreign countries. Other functionality in the financial management modules will include budgets, cash-flow, expense and tax reporting. The evaluation team should focus on areas that are most important to support the strategic plans for your organization.

Business Intelligence: Business Intelligence (BI) has become a standard component of most ERP packages. In general, BI tools allow users to share and analyze the data collected across the enterprise and centralized in the ERP database. BI can come in the form of dashboards, automated reporting and analysis tools used to monitor the organization business performance. BI supports informed decision making by everyone, from executives to line managers and accountants.

Supply Chain Management: Supply Chain Management (SCM), sometimes referred to as logistics, improves the flow of materials through an organization by managing planning, scheduling, procurement, and fulfillment, to maximize customer satisfaction and profitability. Sub modules in SCM often include production scheduling, demand management, distribution management, inventory management, warehouse management, procurement and order management. Any company dealing with products, from manufacturers to distributors, needs to clearly define their SCM requirements to properly evaluate an ERP solution. It easy for a vendor to focus on their applications strengths and not address the full needs of the company.

Human Resource Management: Human resource management ERP modules should enhance the employee experience – from initial recruitment to time tracking. A Sub module can include payroll, performance management, time tracking, benefits, compensation and workforce planning. Self-service tools that allow managers and employees to enter time and attendance, choose benefits and manage PTO are available in many ERP solutions.

Manufacturing Operations: Manufacturing modules make manufacturing operations more efficient through product configuration, job costing and bill of materials management. ERP manufacturing modules often include Capacity Requirements

Planning, Materials Requirements Planning, forecasting, Master Production Scheduling, work-order management and shop-floor control.

Integration: Key to the value of an ERP package is the integration between modules, so that all of the core business functions are connected. Information should flow across the organization so that BI reports on organization-wide results.

ERP can be easier than you imagine – Microsoft Dynamics ERP is cost effective and familiar to your users. If you are thinking about upgrading your systems to a fully integrated ERP system, give us a call.

NEED FOR ERP:

- Real-time information for decisions
- Best practice procedures
- Improved visibility
- Faster month-end close
- Increased customer satisfaction
- Managed and controlled costs
- Better operational efficiency
- Accurate records
- Balance of supply and demand
- Reduced lead times and increased throughput

Real-time information for decisions: Without an ERP system, your team is flying blind. They make decisions based on guesswork and rules of thumb because they don't have the data they need. Sometimes they are the right decisions, but more often, they are sub-optimum decisions that can cost you money and customer goodwill.

Best practice procedures: Software companies often design their ERP systems to support specific industries or verticals. As they add customers, they learn industry best practices and incorporate them into the software. By implementing an ERP system designed for your industry, you automatically make your business processes more efficient.

Improved visibility: If customers want to know when their order will ship or if you need to know whether you have enough of a critical component to accept a rush order, an ERP system gives you instant visibility into your operations and your supply chain.

Faster month-end close: ERP systems automatically process transactions and generate audit trails and financial reports that can simplify period-end closings. They flag anomalies so you can investigate quickly, and they simplify repetitive journal

entries and other activities that make closing so complex and time consuming. Faster closes mean you know the health of your business sooner.

Increased customer satisfaction: Customers like accurate delivery dates, and ERP can help you provide them with improved inventory and shop floor visibility. In addition, the increased visibility and accuracy will help you improve your DIFOT rate (delivery in full on time), which helps keep customers happy.

Managed and controlled costs: ERP systems calculate and collect costs so you always have an accurate picture of your product cost and margins.

Better operational efficiency: By helping you to plan production more effectively, your operational efficiency will improve as you reduce setups and teardowns or unnecessary downtime.

Accurate records: The uniformity of record data that an ERP system instills will help ensure that your records are more accurate, which will increase process accuracy across the board.

Balance of supply and demand: **MRP**, a component of ERP systems, will help you balance supply and demand so you can reduce inventory while keeping customers happy.

Reduced lead times and increased throughput: Better scheduling and accurate records ensure that your schedules focus on priorities, leading to shorter lead times. Since you won't have as many orders waiting for tooling or parts, your throughput will increase.

Advantages of ERP

1. Complete **visibility** into all the important processes, across various departments of an organization.
2. Automatic and coherent **workflow** from one department/function to another, to ensure a smooth transition and quicker completion of processes.
3. A unified and single **reporting** system to analyze the statistics/status etc. in real-time, across all functions/departments.
4. There are **various modules** in an ERP system like Finance/Accounts, Human Resource Management, Manufacturing, Marketing/Sales, Supply Chain/Warehouse Management, CRM, Project Management, etc.
5. ERP systems make it **easier** for order tracking, inventory tracking, revenue tracking, sales forecasting and related activities.

Disadvantages of ERP

1. The **cost** of ERP Software, planning, customization, configuration, testing, implementation, etc. is too high.

2. ERP deployments are highly time-consuming – projects may take **1-3 years** (or more) to get completed and fully functional.
3. Too little **customization** may not integrate the ERP system with the business process & too much customization may slow down the project and make it difficult to upgrade.
4. The **cost savings/payback** may not be realized immediately after the ERP implementation & it is quite difficult to measure the same.

2. Explain in detail about Implementation of ERP and its Functional characteristics

Implementation of ERP

Challenges in implementing ERP solutions are quite normal. Though it is not completely a technical job, a lot of planning and proper communication is very much essential to implement ERP across the organization. Below are the 7 common challenges we have noticed companies experience, when ERP is implemented.

- It is very important, that **implementation is done in stages**. Trying to implement everything at once will lead to a lot of confusion and chaos.
- **Appropriate training is very essential** during and after the implementation. The staff should be comfortable in using the application or else, it will backfire, with redundant work and functional inefficiencies.
- **Lack of proper analysis of requirements** will lead to non-availability of certain essential functionalities. This might affect the operations in the long run and reduce the productivity and profitability.
- **Lack of Support from Senior Management** will lead to unnecessary frustrations in work place. Also, it will cause delay in operations and ineffective decisions. So, it is essential to ensure that the Senior Management supports the transformation.
- **Compatibility Issues with ERP Modules** lead to issues in integration of modules. Companies associate different vendors to implement different ERP modules, based on their competency. It is very essential that there is a way to handle compatibility issues.
- **Cost Overheads** will result, if requirements are not properly discussed and decided during the planning phase. So, before execution, a detailed plan with a complete breakdown of requirements should be worked out.
- **Investment in Infrastructure** is very essential. ERP applications modules will require good processing speed and adequate storage. Not allocating suitable

budget for infrastructure will result in reduced application speed and other software issues. Hardware and Software Security is also equally important.

Characteristics of ERP

ERP (Enterprise Resource Planning) systems typically include the following characteristics:

- An integrated system that operates in (or near) real time without relying on periodic updates
- A common database that supports all applications
- A consistent look and feel across modules
- Installation of the system with elaborate application/data integration by the Information Technology department provided the implementation is not done in small steps.
- Flexible
- Modular and Open
- Comprehensive
- Online-Connection with Other ERP System
- Best Business Practices
- Multi-Facilities
- Strategic planning
- Optimize the data
- Project Management
- Automatic Functions

Flexible: Currently way of doing [business](#) is changing fast due to changing in law and amendments in the standards. So, management of business process which is done through ERP should be flexible. If ERP system will be flexible, we can change processing system as per our requirement.



Modular and Open: One of the great characteristics of any good ERP system that it has open module architecture. It means, if there is error in any module, we can correct

it by opening it instead affecting all other modules. For example, Open ERP has following module which can be open separately.

- Sales Management
- Purchase Management
- Customer Relationship Management
- Project Management
- Accounting & Finance
- Human Resource Management
- Knowledge and Document Management
- Time Tracking
- Manufacturing Resource Planning
- Portal
- Employee Directory
- Address Book
- Recruitment Process



Comprehensive: ERP system should be advance and it should use comprehensive way. It means all most all the functions of business should be done through ERP System. If we will get only small number of activities through ERP and other will be done through manual, then, this ERP system is not ok. Its capacity should to cover all the functions of business.

Online-Connection with Other ERP System: Today, in the market, there are lots of ERP solutions but which is the best, it will tell its features? Check whether it has capacity to connect other ERP system online or not. Because, today business has started to interact with millions of other business. So, it is necessary to connect them online through our ERP.



Best Business Practices: Each business activities have lots of standards. For example, accounting follows IFRS, quality management follows ISO 9000 and marketing follows the standard of MASB. So, your ERP system will updated regarding all standards.

Multi-Facilities: A good ERP System should have multi-facilities. It means, it can work in multi-currencies, multi-mode manufacturing and multi-platform.

Strategic planning: Strategic Planning is the main and top function of business. It should be done through ERP. ERP should integrate all its sub-part systems for making better strategic planning.

Optimize the data: A good ERP system optimize the data for effective utilization of limited business resources. It also optimizes the data for reducing cost and risk.

Project Management: A good ERP System collaborate the team in real time for working together on a project. Everything about the project process can be tracked through this.

Automatic Functions: We can measure the quality of ERP from its advance automatic functions. With these automatic functions, organization saves his lots of time. This automatic function may be in electronic fund transfer, electronic data interchange and ecommerce.

FUNCTIONAL AREAS OF ERP

An ERP system covers the following common functional areas. In many ERP systems these are called and grouped together as ERP modules:

- Financial accounting: General ledger, fixed asset, payables including vouchering, matching and payment, receivables cash application and collections, cash management, financial consolidation.
- Management accounting: Budgeting, costing, cost management, activity based costing.

- Human resources: Recruiting, training, rostering, payroll, benefits, 401K, diversity management, retirement, separation.
- Manufacturing: Engineering, bill of materials, work orders, scheduling, capacity, workflow management, quality control, manufacturing process, manufacturing projects, manufacturing flow, product life cycle management.
- Order Processing: Order to cash, order entry, credit checking, pricing, available to promise, inventory, shipping, sales analysis and reporting, sales commissioning.
- Supply chain management: Supply chain planning, supplier scheduling, product configurator, order to cash, purchasing, inventory, claim processing, warehousing (receiving, putaway, picking and packing).
- Project management: Project planning, resource planning, project costing, work breakdown structure, billing, time and expense, performance units, activity management.
- Customer relationship management: Sales and marketing, commissions, service, customer contact, call center support — CRM systems are not always considered part of ERP systems but rather Business Support systems (BSS).
- Data Services: Various "self-service" interfaces for customers, suppliers and/or employees.

3. Explain in detail about ERP System Architecture

ERP SYSTEM ARCHITECTURE: Enterprise resource planning (ERP) is business-management software typically a suite of integrated applications that an organization can use to collect, store, manage and interpret data from many business activities, including:

- Product planning, cost
- Manufacturing or service delivery
- Marketing and sales
- Inventory management
- Shipping and payment

ERP provides an integrated view of core business processes, often in real-time, using common databases maintained by a database management system. ERP systems track business resources such as cash, raw materials, production capacity and the status of business commitments: orders, purchase orders, and payroll. The applications that make up the system share data across various departments (manufacturing, purchasing, sales, accounting, etc.) that provide the data. ERP

facilitates information flow between all business functions, and manages connections to outside stakeholders.

The ERP system is considered a vital organizational tool because it integrates varied organizational systems and facilitates error-free transactions and production. However, developing an ERP system differs from traditional system development. ERP systems run on a variety of computer hardware and network configurations, typically using a database as an information repository.

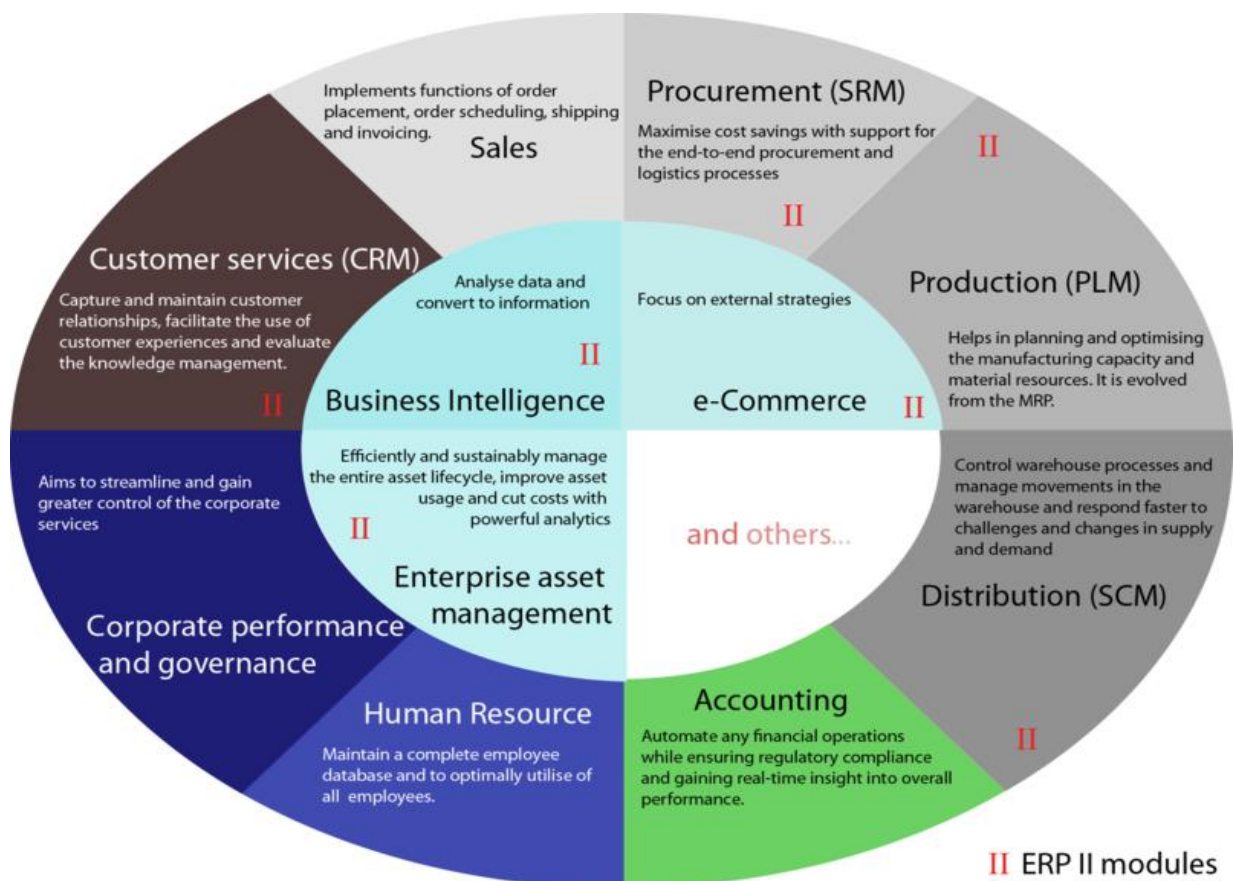
ERP modules:

The key role of an ERP system is to provide support for such business functions as accounting, sales, inventory control, and production.

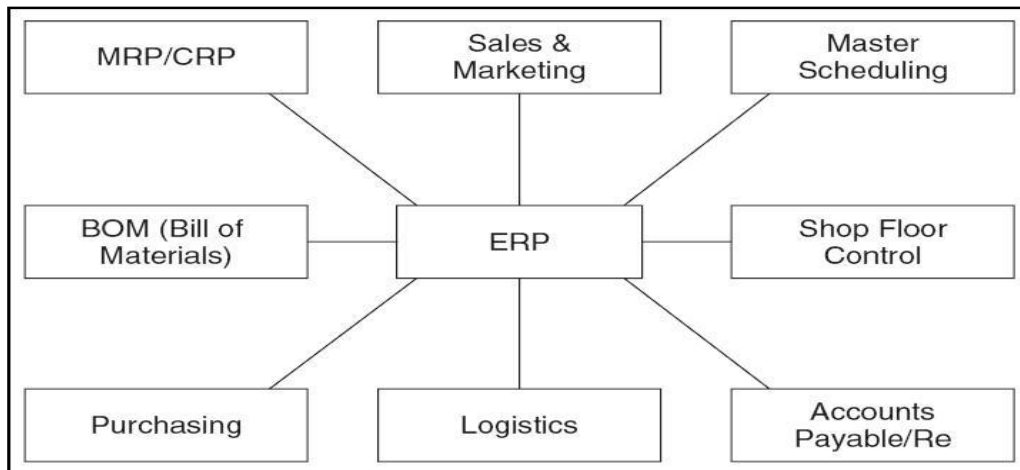
ERP vendors, including SAP, Oracle, and Microsoft, etc. provide modules that support the major functional areas of a business.

The ERP software embeds best business practices that implement the organization's policy and procedure via business rules.

Typical ERP Modules:



Typical ERP Module



ERP applications are most commonly deployed in a distributed and often widely dispersed manner. While the servers may be centralized, the clients are usually spread to multiple locations throughout the enterprise.

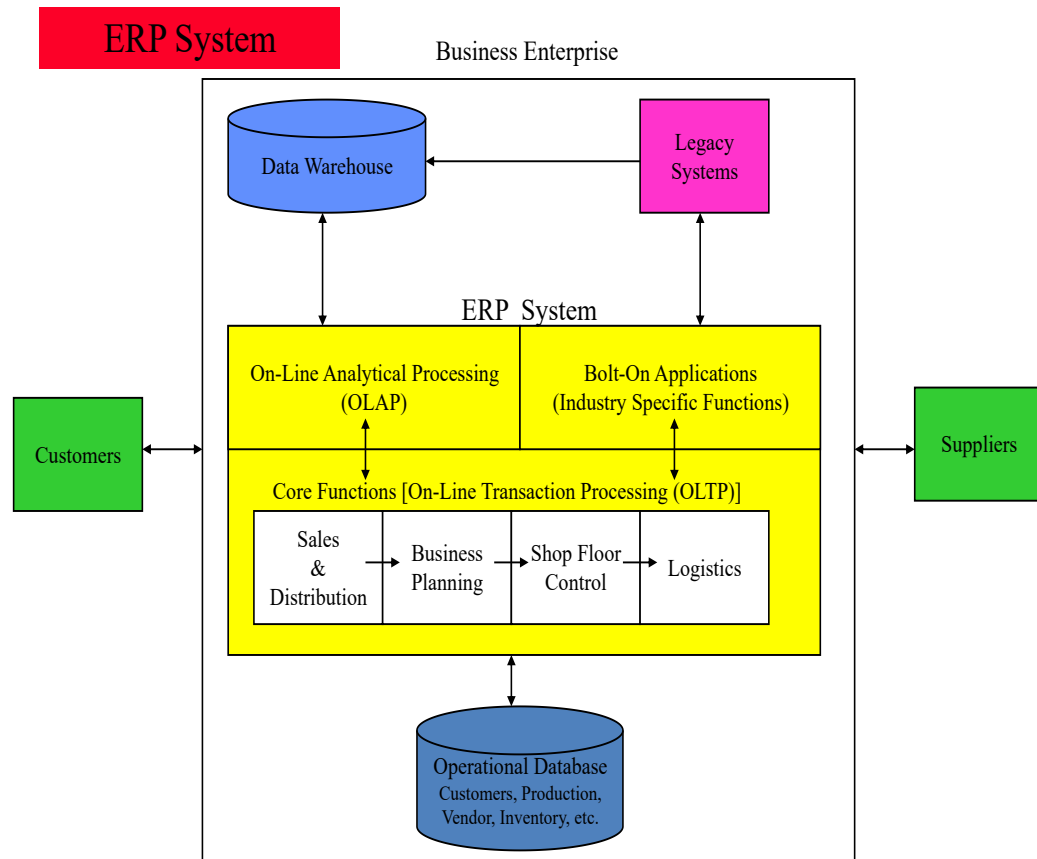
Generally there are three functional areas of responsibility that is distributed among the servers and the clients. First, there is the database component - the central repository for all of the data that is transferred to and from the clients. Then, of course, the clients - here raw data gets inputted, requests for information are submitted, and the data satisfying these requests is presented. Lastly, we have the application component that acts as the intermediary between the client and the database. Where these components physically reside and how the processes get distributed will vary somewhat from one implementation to the next. The two most commonly implemented architectures are outlined below.

Two-tier Implementations

In typical two-tier architecture, the server handles both application and database duties. The clients are responsible for presenting the data and passing user input back to the server. While there may be multiple servers and the clients may be distributed across several types of local and wide area links, this distribution of processing responsibilities remains the same.

Three-tier Client/Server Implementations

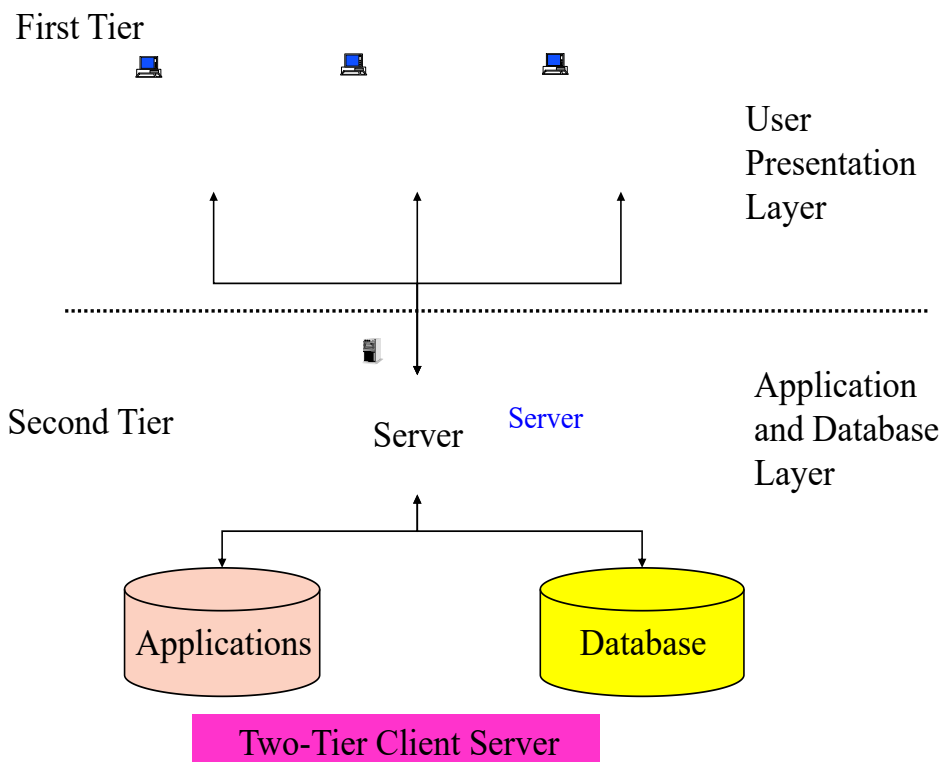
In three-tier architectures, the database and application functions are separated. This is very typical of large production ERP deployments. In this scenario, satisfying client requests requires two or more network connections. Initially, the client establishes communications with the application server. The application server then creates a second connection to the database server.



ERP System Configurations: Client-Server Network Topology

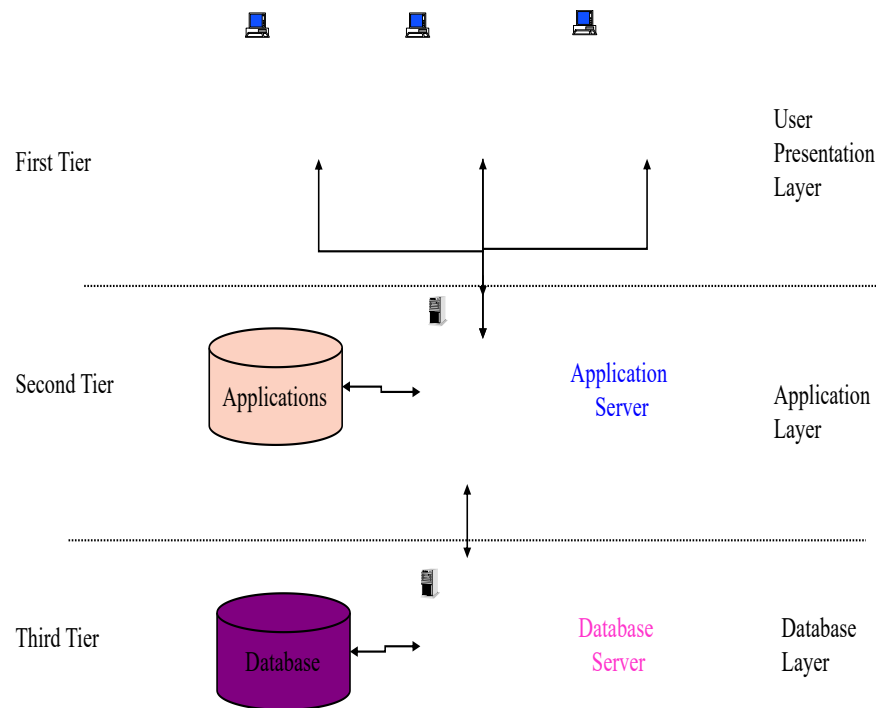
Two-tier:

- It common server handles both application and database duties.
- It is used especially in LANs.



Three-tier:

- The client links to the application server which then initiates a second connection to the database server.
- It is used especially in WANs.



Three-Tier Client Server

4. Write Short notes on advantages and disadvantages of ERP.**Benefits of ERP:**

- ERP can improve quality and efficiency of the business. By keeping a company's internal business processes running smoothly, ERP can lead to better outputs that may benefit the company, such as in customer service and manufacturing.
- ERP supports upper level management by providing information for decision making.
- ERP creates a more agile company that adapts better to change. ERP makes a company more flexible and less rigidly structured so organization components operate more cohesively, enhancing the business—internally and externally.
- ERP can improve data security. A common control system, such as the kind offered by ERP systems, allows organizations the ability to more easily ensure key company data is not compromised.

- ERP provides increased opportunities for collaboration. Data takes many forms in the modern enterprise. Documents, files, forms, audio and video, emails. Often, each data medium has its own mechanism for allowing collaboration. ERP provides a collaborative platform that lets employees spend more time collaborating on content rather than mastering the learning curve of communicating in various formats across distributed systems.
- Large numbers of end-users have access to ERP applications over the Web.
- Easily integrate ERP applications with existing systems.
- Server-centric—No complex, expensive client software installation.
- The server-centric architecture enables secure end-user access to ERP applications.
- Client-centric—Architecture has better response time because user requests are mostly processed on the client's computer.
- Web-based architectures also allow better system-to-system integration.

Disadvantages of ERP:

- Customization can be problematic.
- Re-engineering business processes to fit the ERP system may damage competitiveness or divert focus from other critical activities.
- ERP can cost more than less integrated or less comprehensive solutions.
- High ERP switching costs can increase the ERP vendor's negotiating power, which can increase support, maintenance, and upgrade expenses.
- Overcoming resistance to sharing sensitive information between departments can divert management attention.
- Integration of truly independent businesses can create unnecessary dependencies.
- Extensive training requirements take resources from daily operations.
- Due to ERP's architecture (OLTP, On-Line Transaction Processing) ERP systems are not well suited for production planning and supply chain management (SCM).
- Harmonization of ERP systems can be a mammoth task (especially for big companies) and requires a lot of time, planning, and money.

5. Explain in Detail about Business Process Re-engineering.

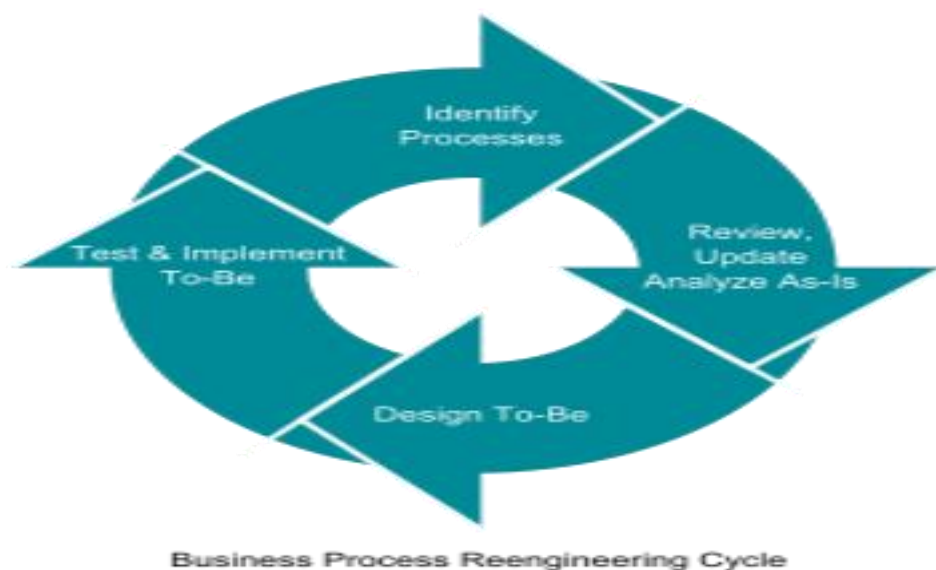
Business process re-engineering: Business process re-engineering (BPR) is the analysis and redesign of workflows within and between enterprises in order to optimize end-to-end processes and automate non-value-added tasks. Business Process Reengineering (BPR) is the practice of rethinking and redesigning the way work is done to better support an organization's mission and reduce costs. Reengineering

starts with a high-level assessment of the organization's mission, strategic goals, and customer needs.

Within the framework of this basic assessment of mission and goals, re-engineering focuses on the organization's business processes—the steps and procedures that govern how resources are used to create products and services that meet the needs of particular customers or markets. As a structured ordering of work steps across time and place, a business process can be decomposed into specific activities, measured, modeled, and improved. It can also be completely redesigned or eliminated altogether. Re-engineering identifies, analyzes, and re-designs an organization's core business processes with the aim of achieving dramatic improvements in critical performance measures, such as cost, quality, service, and speed.

Re-engineering recognizes that an organization's business processes are usually fragmented into sub processes and tasks that are carried out by several specialized functional areas within the organization. Re-engineering maintains that optimizing the performance of sub processes can result in some benefits, but cannot yield dramatic improvements if the process itself is fundamentally inefficient and outmoded.

For that reason, re-engineering focuses on re-designing the process as a whole in order to achieve the greatest possible benefits to the organization and their customers. This drive for realizing dramatic improvements by fundamentally re-thinking how the organization's work should be done distinguishes the re-engineering from process improvement efforts that focus on functional or incremental improvement. Fig Shows business process reengineering cycle.



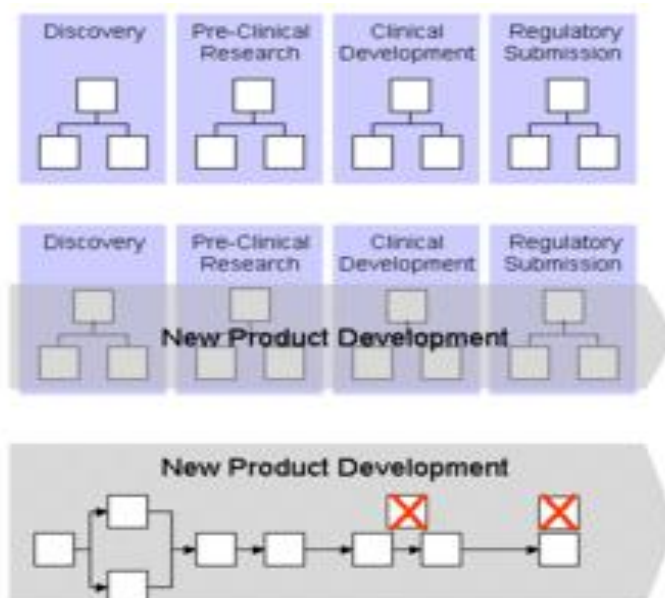


Reengineering guidance and relationship of Mission and Work Processes to Information Technology.

The most notable definitions of reengineering are:

- The fundamental rethinking and radical redesign of business processes to achieve dramatic improvements in critical contemporary modern measures of performance, such as cost, quality, service, and speed.
- Encompasses the envisioning of new work strategies, the actual process design activity, and the implementation of the change in all its complex technological, human, and organizational dimensions."

Research and methodology:



Model based on PRLC approach:

Although the labels and steps differ slightly, the early methodologies that were rooted in IT-centric BPR solutions share many of the same basic principles and elements. The following outline is one such model, based on the PRLC (Process Reengineering Life Cycle). Simplified schematic outline of using a business process approach, exemplified for pharmaceutical R&D.

- Structural organization with functional units
- Introduction of New Product Development as cross-functional process
- Re-structuring and streamlining activities, removal of non-value adding tasks

BPR Success & Failure Factors

Factors that are important to BPR success include:

1. BPR team composition.
2. Business needs analysis.
3. Adequate IT infrastructure.
4. Effective change management.
5. Ongoing continuous improvement

There are many reasons for sub-optimal business processes which include:

1. One department may be optimized at the expense of another
2. Lack of time to focus on improving business process
3. Lack of recognition of the extent of the problem
4. Lack of training
5. People involved use the best tool they have at their disposal which is usually Excel to fix problems
6. Inadequate infrastructure
7. Overly bureaucratic processes
8. Lack of motivation

Organization wide commitment:

Major changes to business processes have a direct impact on processes, technology, job roles, and workplace culture. Significant changes to even one of those areas require resources, money, and leadership. Changing them simultaneously is an extraordinary task. Like any large and complex undertaking, implementing reengineering requires the talents and energies of a broad spectrum of experts. Since BPR can involve multiple areas within the organization, it is important to get support from all affected departments. Through the involvement of selected department members, the organization can gain valuable input before a process is implemented; a step which promotes both the cooperation and the vital acceptance of the reengineered process by all segments of the organization.

Getting enterprise wide commitment involves the following: top management sponsorship, bottom-up buy-in from process users, dedicated BPR team, and budget allocation for the total solution with measures to demonstrate value. Before any BPR project can be implemented successfully, there must be a commitment to the project by the management of the organization, and strong leadership must be

provided. Reengineering efforts can by no means be exercised without a company-wide commitment to the goals. However, top management commitment is imperative for success. Top management must recognize the need for change, develop a complete understanding of what BPR is, and plan how to achieve it.

Leadership has to be effective, strong, visible, and creative in thinking and understanding in order to provide a clear vision. Convincing every affected group within the organization of the need for BPR is a key step in successfully implementing a process. By informing all affected groups at every stage, and emphasizing the positive end results of the reengineering process, it is possible to minimize resistance to change and increase the odds for success.

The ultimate success of BPR depends on the strong, consistent, and continuous involvement of all departmental levels within the organization.

It also depends on the people who do it and how well they can be motivated to be creative and to apply their detailed knowledge to the redesign of business processes.

BPR team composition:

Once organization-wide commitment has been secured from all departments involved in the reengineering effort and at different levels, the critical step of selecting a BPR team must be taken.

This team will form the nucleus of the BPR effort, make key decisions and recommendations, and help communicate the details and benefits of the BPR program to the entire organization. The determinants of an effective BPR team may be summarized as follows:

- competency of the members of the team, their motivation,
- their credibility within the organization and their creativity,
- team empowerment, training of members in process mapping and brainstorming techniques,
- effective team leadership,
- Proper organization of the team,
- Complementary skills among team members, adequate size, interchangeable accountability, clarity of work approach, and
- Specificity of goals.

The most effective BPR teams include active representatives from the following work groups: top management, business area responsible for the process being addressed, technology groups, finance, and members of all ultimate process users' groups.

Team members who are selected from each work group within the organization will have an impact on the outcome of the reengineered process according to their desired requirements. The BPR team should be mixed in depth and knowledge. For example, it may include members with the following characteristics:

- Members who do not know the process at all.
- Members who know the process inside-out.
- Customers, if possible.
- Members representing impacted departments.
- One or two members of the best, brightest, passionate, and committed technology experts.
- Members from outside of the organization

Business needs analysis:

This plan includes the following:

- Identifying specific problem areas,
- Solidifying particular goals, and
- Defining business objectives.

The business needs analysis contributes tremendously to the re-engineering effort by helping the BPR team to prioritize and determine where it should focus its improvements efforts.

The business needs analysis also helps in relating the BPR project goals back to key business objectives and the overall strategic direction for the organization. This linkage should show the thread from the top to the bottom of the organization, so each person can easily connect the overall business direction with the re-engineering effort.

This alignment must be demonstrated from the perspective of financial performance, customer service, associate value, and the vision for the organization.

Developing a business vision and process objectives relies, on the one hand, on a clear understanding of organizational strengths, weaknesses, and market structure, and on the other, on awareness and knowledge about innovative activities undertaken by competitors and other organizations.

6. Explain in Detail about Management Information System (MIS)

Management Information System (MIS):

- A **management information system (MIS)** is a computerized database of financial **information** organized and programmed in such a way that it produces regular reports on operations for every level of **management** in a company.
- It is usually also possible to obtain special reports from the **system** easily.

Types and terminology:

The terms management information system (MIS), information system, enterprise resource planning (ERP), and information technology management are often confused. Information systems and MIS are broader categories that include ERP. Information technology management concerns the operation and organization of information technology resources independent of their purpose. Management information systems, produce fixed, regularly scheduled reports based on data extracted and summarized from the firm's underlying transaction processing systems to middle and operational level managers to identify and inform structured and semi-structured decision problems.

Decision support systems (DSS) are computer program applications used by middle and higher management to compile information from a wide range of sources to support problem solving and decision making. A DSS is used mostly for semi-structured and unstructured decision problems.

Executive information systems (EIS) is a reporting tool that provides quick access to summarized reports coming from all company levels and departments such as accounting, human resources and operations.

Marketing Information Systems are Management Information Systems designed specifically for managing the marketing aspects of the business

Accounting information systems are focused accounting functions.

Human resource management systems are used for personnel aspects.

Office automation systems (OAS) support communication and productivity in the enterprise by automating workflow and eliminating bottlenecks. OAS may be implemented at any and all levels of management.

School Information Management Systems (SIMS) cover school administration and often including teaching and learning materials.

Enterprise resource planning facilitates the flow of information between all business functions inside the boundaries of the organization and manage the connections to outside stakeholders.

Advantages:

The following are some of the benefits that can be attained using MISs.

Companies are able to identify their strengths and weaknesses due to the presence of revenue reports, employees' performance record etc. Identifying these aspects can help a company improve its business processes and operations.

Giving an overall picture of the company.

Acting as a communication and planning tool.

The availability of customer data and feedback can help the company to align its business processes according to the needs of its customers. The effective management of customer data can help the company to perform direct marketing and promotion activities.

MISs can help a company gain a competitive advantage. Competitive advantage is a firm's ability to do something better, faster, cheaper, or uniquely, when compared with rival firms in the market.

Enterprise applications:

Enterprise systems—also known as enterprise resource planning (ERP) systems—provide integrated software modules and a unified database that personnel use to plan, manage, and control core business processes across multiple locations. Modules of ERP systems may include finance, accounting, marketing, human resources, production, inventory management, and distribution.

Supply chain management (SCM) systems enable more efficient management of the supply chain by integrating the links in a supply chain. This may include suppliers, manufacturers, wholesalers, retailers, and final customers.

Customer relationship management (CRM) systems help businesses manage relationships with potential and current customers and business partners across marketing, sales, and service.

Knowledge management system (KMS) helps organizations facilitate the collection, recording, organization, retrieval, and dissemination of knowledge. This may include documents, accounting records, unrecorded procedures, practices, and skills.

Knowledge management (KM) as a system covers the process of knowledge creation and acquisition from internal processes and the external world. The collected knowledge is incorporated in organizational policies and procedures, and then disseminated to the stakeholders.

7.Explain in Detail about Decision support system (DSS)

Decision support system (DSS) is a computer-based information system that supports business or organizational decision-making activities. DSSs serve the management, operations, and planning levels of an organization (usually mid and higher management) and help people make decisions about problems that may be rapidly changing and not easily specified in advance—i.e. Unstructured and Semi-Structured decision problems. Decision support systems can be either fully computerized, human-powered or a combination of both.

Typical information that a decision support application might gather and present includes:

- Inventories of information assets (including legacy and relational data sources, cubes, data warehouses, and data marts),
- Comparative sales figures between one period and the next,
- Projected revenue figures based on product sales assumptions.

DSSs are often contrasted with more automated decision-making systems known as Decision Management Systems.

Classification: DSS components may be classified as:

-
- **Inputs:** Factors, numbers, and characteristics to analyze
 - **User Knowledge and Expertise:** Inputs requiring manual analysis by the user
 - **Outputs:** Transformed data from which DSS "decisions" are generated
 - **Decisions:** Results generated by the DSS based on user criteria

Development frameworks:

DSS systems are not entirely different from other systems and require a structured approach. Such a framework includes people, technology, and the development approach. The Early Framework of Decision Support System consists of four phases:

- **Intelligence** Searching for conditions that call for decision.
- **Design** Developing and analyzing possible alternative actions of solution.
- **Choice** Selecting a course of action among those.
- **Implementation** Adopting the selected course of action in decision situation.

DSS technology levels (of hardware and software) may include:

- The actual application that will be used by the user. This is the part of the application that allows the decision maker to make decisions in a particular problem area. The user can act upon that particular problem.
- Generator contains Hardware/software environment that allows people to easily develop specific DSS applications. This level makes use of case tools or systems such as Crystal, Analytica and iThink.
- Tools include lower level hardware/software. DSS generators including special languages, function libraries and linking modules
- An iterative developmental approach allows for the DSS to be changed and redesigned at various intervals. Once the system is designed, it will need to be tested and revised where necessary for the desired outcome.

8. Explain in Detail about Executive Information System/Executive Support Systems

EXECUTIVE INFORMATION SYSTEM/EXECUTIVE SUPPORT SYSTEMS

An **executive information system (EIS)**, also known as an **executive support system (ESS)**, is a type of management information system that facilitates and supports senior executive information and decision-making needs. It provides easy access to internal and external information relevant to organizational goals. It is commonly considered a specialized form of decision support system (DSS). EIS emphasizes graphical displays and easy-to-use user interfaces.

Components: EIS components can typically be classified as:

-
- Hardware
 - Software
 - User interface
 - Telecommunications

Hardware:

The basic hardware needed for a typical EIS includes four components:

1. Input data-entry devices. These devices allow the executive to enter, verify, and update data immediately
2. The central processing unit (CPU), which is the important because it controls the other computer system components
3. Data storage files. The executive can use this part to save useful business information, and this part also help the executive to search historical business information easily
4. Output devices, which provide a visual or permanent record for the executive to save or read. This device refers to the visual output device such as monitor or printer

Software:

Choosing the appropriate software is vital to an effective therefore, the software components and how they integrate the data into one system are important. A typical EIS includes four software components:

1. Text-handling software—documents are typically text-based
2. Database—heterogeneous databases on a range of vendor-specific and open computer platforms help executives access both internal and external data
3. Graphic base—graphics can turn volumes of text and statistics into visual information for executives. Typical graphic types are: time series charts, scatter diagrams, maps, motion graphics, sequence charts, and comparison-oriented graphs (i.e., bar charts)
4. Model base—EIS models contain routine and special statistical, financial, and other quantitative analysis

User interface:

An EIS must be efficient to retrieve relevant data for decision makers, so the user interface is very important. Several types of interfaces can be available to the EIS structure, such as scheduled reports, questions/answers, menu driven, command language, natural language, and input/output.

Telecommunication:

As decentralizing is becoming the current trend in companies, telecommunications will play a pivotal role in networked information systems.

Transmitting data from one place to another has become crucial for establishing a reliable network. In addition, telecommunications within an EIS can accelerate the need for access to distributed data.

Applications:

-
- Manufacturing
 - Marketing
 - Financial

Advantages of EIS:

- Easy for upper-level executives to use, extensive computer experience is not required in operations.
- Provides timely delivery of company summary information
- Information that is provided is better understood
- EIS provides timely delivery of information. Management can make decisions promptly.
- Improves tracking information
- Offers efficiency to decision makers

Disadvantages of EIS:

- System dependent
- Limited functionality, by design
- Information overload for some managers
- Benefits hard to quantify
- High implementation costs
- System may become slow, large, and hard to manage
- Need good internal processes for data management
- May lead to less reliable and less secure data

9. Explain in Detail about OLAP (Online Analytical Processing).

ON -LINE ANALYTICAL PROCESSING:

- **OLAP (Online Analytical Processing)** is the technology behind many Business Intelligence (BI) applications. OLAP is a powerful technology for data discovery, including capabilities for limitless report viewing, complex analytical calculations, and predictive “what if” scenario (budget, forecast) planning.

Multidimensional:

MOLAP (multi-dimensional online analytical processing) is the classic form of OLAP and is sometimes referred to as just OLAP. MOLAP stores this data in an optimized multi-dimensional array storage, rather than in a relational database.

Some MOLAP tools require the pre-computation and storage of derived data, such as consolidations - the operation known as processing. Such MOLAP tools generally utilize a pre-calculated data set referred to as a data cube. The data cube contains all the possible answers to a given range of questions. As a result, they have a very fast response to queries.

On the other hand, updating can take a long time depending on the degree of pre-computation. Pre-computation can also lead to what is known as data explosion. Other MOLAP tools, particularly those that implement the functional database model do not pre-compute derived data but make all calculations on demand other than those that were previously requested and stored in a cache.

Advantages of MOLAP

- Fast query performance due to optimized storage, multidimensional indexing and caching.
- Smaller on-disk size of data compared to data stored in relational database due to compression techniques.
- Automated computation of higher level aggregates of the data.
- It is very compact for low dimension data sets.
- Array models provide natural indexing.
- Effective data extraction achieved through the pre-structuring of aggregated data.

Disadvantages of MOLAP

- Within some MOLAP Solutions the processing step (data load) can be quite lengthy, especially on large data volumes. This is usually remedied by doing only incremental processing, i.e., processing only the data which have changed (usually new data) instead of reprocessing the entire data set.
- Some MOLAP methodologies introduce data redundancy.

Relational:

ROLAP works directly with relational databases. The base data and the dimension tables are stored as relational tables and new tables are created to hold the aggregated information. It depends on a specialized schema design. This methodology relies on manipulating the data stored in the relational database to give the appearance of traditional OLAP's slicing and dicing functionality. In essence, each action of slicing and dicing is equivalent to adding a "WHERE" clause in the SQL statement. ROLAP tools do not use pre-calculated data cubes but instead pose the query to the standard relational database and its tables in order to bring back the data required to answer the question.

ROLAP tools feature the ability to ask any question because the methodology does not limit to the contents of a cube. ROLAP also has the ability to drill down to the lowest level of detail in the database.

10. Explain in Detail about Supply chain management (SCM).**SUPPLY CHAIN MANAGEMENT (SCM):**

Supply chain management (SCM) is the **management** of the flow of goods and services.

It includes the movement and storage of raw materials, work-in-process inventory, and finished goods from point of origin to point of consumption.

Business process integration:

Successful SCM requires a change from managing individual functions to integrating activities into key supply chain processes. In an example scenario, a purchasing department places orders as its requirements become known. The marketing department, responding to customer demand, communicates with several distributors and retailers as it attempts to determine ways to satisfy this demand. Information shared between supply chain partners can only be fully leveraged through process integration.

Supply chain business process integration involves collaborative work between buyers and suppliers, joint product development, common systems, and shared information. According to Lambert and Cooper (2000), operating an integrated supply chain requires a continuous information flow. However, in many companies, management has concluded that optimizing product flows cannot be accomplished without implementing a process approach. The key supply chain processes stated by Lambert are:

- Customer relationship management
- Customer service management

- Demand management style
- Order fulfillment
- Manufacturing flow management
- Supplier relationship management
- Product development and commercialization
- Returns management

Customer service management process:

Customer relationship management concerns the relationship between an organization and its customers. Customer service is the source of customer information. It also provides the customer with real-time information on scheduling and product availability through interfaces with the company's production and distribution operations. Successful organizations use the following steps to build customer relationships:

- Determine mutually satisfying goals for organization and customers
- Establish and maintain customer rapport
- Induce positive feelings in the organization and the customer

Procurement process:

Strategic plans are drawn up with suppliers to support the manufacturing flow management process and the development of new products. In firms whose operations extend globally, sourcing may be managed on a global basis.

The desired outcome is a relationship where both parties benefit and a reduction in the time required for the product's design and development. The purchasing function may also develop rapid communication systems, such as electronic data interchange (EDI) and Internet linkage, to convey possible requirements more rapidly.

Product development and commercialization:

Here, customers and suppliers must be integrated into the product development process in order to reduce the time to market.

As product life cycles shorten, the appropriate products must be developed and successfully launched with ever-shorter time schedules in order for firms to remain competitive.

According to Lambert and Cooper (2000), managers of the product development and commercialization process must:

1. Coordinate with customer relationship management to identify customer-articulated needs;
2. Select materials and suppliers in conjunction with procurement; and
3. Develop production technology in manufacturing flow to manufacture and integrate into the best supply chain flow for the given combination of product and markets.

Manufacturing flow management process:

The manufacturing process produces and supplies products to the distribution channels based on past forecasts.

Manufacturing processes must be flexible in order to respond to market changes and must accommodate mass customization.

Orders are processes operating on a just-in-time (JIT) basis in minimum lot sizes. Changes in the manufacturing flow process lead to shorter cycle times, meaning improved responsiveness and efficiency in meeting customer demand.

Physical distribution:

This concerns the movement of a finished product or service to customers.

In physical distribution, the customer is the final destination of a marketing channel, and the availability of the product or service is a vital part of each channel participant's marketing effort.

It is also through the physical distribution process that the time and space of customer service become an integral part of marketing.

Thus it links a marketing channel with its customers (i.e., it links manufacturers, wholesalers, and retailers).

Performance measurement:

Experts found a strong relationship from the largest arcs of supplier and customer integration to market share and profitability.

Taking advantage of supplier capabilities and emphasizing a long-term supply chain perspective in customer relationships can both be correlated with a firm's performance.

As logistics competency becomes a critical factor in creating and maintaining competitive advantage, measuring logistics performance becomes increasingly important, because the difference between profitable and unprofitable operations becomes narrower.

Warehousing management:

To reduce a company's cost and expenses, warehousing management is concerned with storage, reducing manpower cost, dispatching authority with on time delivery, loading & unloading facilities with proper area, stock management system etc.

Workflow management:

Integrating suppliers and customers tightly into a workflow (or business process) and thereby achieving an efficient and effective supply chain is a key goal of workflow management.

11. Explain in Detail about Customer relationship management (CRM)

CUSTOMER RELATIONSHIP MANAGEMENT (CRM):

Customer relationship management (CRM) is an approach to managing a company's interaction with current and future customers. It often involves using technology to organize, automate, and synchronize sales, marketing, customer service, and technical support.



Characteristics

- Sales force automation, which implements sales promotion analysis, automates the tracking of a client's account history for repeated sales or future sales, and coordinates sales, marketing, call centers, and retail outlets.
- Data warehouse technology, used to aggregate transaction information, to merge the information with CRM products, and to provide key performance indicators.
- Opportunity management which helps the company to manage unpredictable growth and demand, and implement a good forecasting model to integrate sales history with sales.
- CRM systems that track and measure marketing campaigns over multiple networks, tracking customer analysis by customer clicks and sales.

Types:

Call centers

As well as tracking, recording and storing customer information, CRM systems in call centers codify the interactions between company and customers by using analytics and key performance indicators to give the users information on where to focus their marketing and customer service.

The intention is to maximize average revenue per user, decrease churn rate and decrease idle and unproductive contact with the customers. CRM software can also be used to identify and reward loyal customers over a period of time.

Growing in popularity is the idea of gamifying customer service environments. The repetitive and tedious act of answering support calls all day can be draining, even for

the most enthusiastic customer service representative. When agents are bored with their work, they become less engaged and less motivated to do their jobs well. They are also prone to making mistakes. Gamification tools can motivate agents by tapping into their visceral need for reward, status, achievement, and competition.

Business-to-business

According to a Sweeney Group definition, CRM is "all the tools, technologies and procedures to manage, improve, or facilitate sales, support and related interactions with customers, prospects, and business partners throughout the enterprise". It assumes that CRM is involved in every B2B transaction. Despite the general notion that CRM systems were created for the customer-centric businesses, they can also be applied to B2B environments to streamline and improve customer management conditions.

For the best level of CRM operation in a B2B environment, the software must be personalized and delivered at individual levels.

The main differences between B2C and B2B CRM systems are as follows

- B2B companies have smaller contact databases than B2C.
- The volume of sales in B2B is relatively small.
- In B2B there are less figure propositions, but in some cases they cost a lot more than B2C items.
- Relationships in B2B environment are built over a longer period of time.
- B2B CRM must be easily integrated with products from other companies. Such integration enables the creation of forecasts about customer behavior based on their buying history, bills, business success, etc.
- An application for a B2B company must have a function to connect all the contacts, processes and deals among the customers segment and then prepare a paper.
- Automation of sales process is an important requirement for B2B products. It should effectively manage the deal and progress it through all the phases towards signing.
- A crucial point is personalization. It helps the B2B Company to create and maintain strong and long-lasting relationship with the customer.

UNIT II

SAP: History – SAP R/2 – SAP R/3 – Characteristics of SAP R/3 – Architecture of SAP R/3 – SAP Modules, Net Weaver, Customer Relationship Management, Business Warehouse, Advanced Planner and Optimiser. **ABAP/4: Workbench** - Workbench Tools - ABAP/4 Data Dictionary - ABAP/4 Repository Information – Structure of ABAP/4 program - ABAP/4 syntax – Data types – Constants and Variables – Statements : DATA, PARAMETERS, TABLE, MOVE, MOVE-CORRESPONDING, CLEAR, WRITE, CHECK, FORMAT. LOOP STRUCTURES. Sample programs.

TWO MARK QUESTIONS:**1. Define SAP.**

SAP (Systems, Applications & Products in Data Processing) is a European multinational software corporation that makes enterprise software to manage business operations and customer relations

2. List out the Various SAP Modules:

- Financial Accounting (FI)
- Financial Supply Chain Management (FSCM)
- Controlling (CO)
- Materials Management (MM)
- Sales and Distribution (SD)
- Logistics Execution (LE)
- Production Planning (PP)
- Quality Management (QM)
- Plant Maintenance (PM)
- Project System (PS)
- Human Resources (HR)

3. What is CRM?

Customer relationship management (CRM) is an approach to managing a company's interaction with current and future customers. It often involves using technology to organize, automate, and synchronize sales, marketing, customer service, and technical support.

4. Define Business Warehouse

Business Warehouse (BW) is a combination of databases and database management tools that are used to support management decision making.

5. Define SAP Netweaver

NetWeaver is a combination of the underlying SAP Kernel (also known as the SAP OS layer, basically the WEB AS) and any SAP software tool for business enablement.

6. Define Planning & Optimiser

APO (Advanced Planning and Optimization) application is at the heart of SCM. It offers planning and optimization functionalities in different business processes of Demand Planning, Supply Planning, Supply and Demand Matching, Production Planning Detailed Scheduling, and Global Available to Promise and Transportation Management.

7. What is ABAP?

ABAP – the programming language used in SAP for developing business application support and development. ABAP is a programming language that runs in the SAP ABAP runtime environment, created and used by SAP for the development of application programs

8. Define ABAP Data Dictionary?

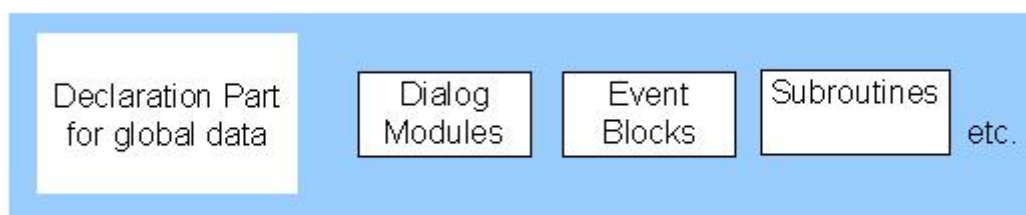
ABAP Dictionary is one of the important tools of ABAP Workbench. It is used to create and manage data definitions without redundancies. ABAP Dictionary always provides the updated information of an object to all the system components. The presence and role of ABAP Dictionary ensures that data stored in an SAP system is consistent and secure.

9. Define Repository Information System

ABAP/4 Repository Information System (in short: Repository Information System) given here is restricted to aspects of relevance for work with the Data Modeler. Using the ABAP/4 Repository Information System, you can search for all modeling objects of the Data Modeler that you wish to display or edit.

10. Define the Structure of ABAP/4 Program

The following diagram shows the structure of an ABAP program:



ELEVEN MARK QUESTIONS:**1. Explain in detail about History of SAP.****SAP-History**

SAP (Systems, Applications & Products in Data Processing) is a European multinational software corporation that makes enterprise software to manage business operations and customer relations. SAP is headquartered in Walldorf, Baden-Württemberg, Germany, with regional offices in 130 countries. The company has over 293,500 customers in 190 countries. The company is a component of the Euro Stoxx 50 stock market index. SAP is the world leader in enterprise applications in terms of software and software-related service revenue. Based on market capitalization, it is the world's third largest independent software manufacturer supporting all sizes of industries helping them to operate profitability, grow sustainably and stay ahead of the competition in the market.

Company Perspectives: Whether opening up new markets or servicing existing ones, SAP is committed to its tested strategy: concentrating on its core business developing and selling standard enterprise applications software and partnering with hardware vendors, software suppliers, technology providers, value-added resellers, and consulting firms.

Company History:

SAP AG is the fifth largest independent software producer worldwide and the largest producer of standard enterprise-wide business applications for the roughly \$9 billion global client-server software market. The company's principal business activities are the development and marketing of an integrated line of prepackaged computer software for over 1,000 predefined business processes, from financial accounting, human resources, and plant maintenance to quality assurance, materials management, sales and distribution, and business workflow. Its two major products, the R/2 and R/3 suite of business software applications, are used by over 4,000 companies in the oil and gas, banking, insurance, utilities, telecommunications, pharmaceuticals, consumer products, automotive, retail, health care, chemicals, and high tech and electronics industries. Through its 28 international subsidiaries, led by its U.S. subsidiary SAP America, SAP AG markets its software and consulting, training, and support services in more than 50 countries.

1970s Founding

SAP AG was founded in 1972 by five German engineers with IBM in Mannheim, Germany; interestingly four of the founders Hasso Plattner, Dietmar Hopp, Klaus Tschira, and Hans Werner Hector were still with SAP in early 1996. When an

IBM client asked IBM to provide enterprise-wide software to run on its mainframe, the five began writing the program only to be told the assignment was being transferred to another unit. Rather than abandon the project altogether, they left IBM and founded SAP in Walldorf, near Heidelberg. SAP eventually came to stand for Systeme, Anwendungen, und Produkte in Datenverarbeitung (systems, applications, and products in data processing).

Without the benefit of loans from banks, venture capitalists, or the German government, SAP began fashioning its software business gradually through the cash flow generated by an ever-growing stable of customers. Working at night on borrowed computers to land their first contracts, Plattner and colleagues built SAP's client list with German firms in its region, beginning with a German subsidiary of the global chemical company ICI and later adding such major German multinationals as Siemens and BMW.

R/2 in the Late 1970s

In 1978 SAP began developing, and the following year released, R/2 (R for "real-time"), a mainframe-based, standard business software suite in which integratable modules for accounting, sales and distribution, and production enabled customers to consolidate their financial and operational data into a single database and eliminate costly paperwork and data entry. Because the modules were self-standing, businesses could select only those they needed, which could then be further customized to their unique requirements. The promise of real-time integration of mission-critical corporate data, viewable through the spreadsheet-like windows of specialized software, offered the potential for uniform data flow, streamlined business operations, and centralized decision-making.

Relying on word of mouth filtering through the overseas branches of its German customers, SAP soon began selling its software outside Europe. With corporate giants like Dow Chemical and Bayer already running R/2, SAP could rely on the fear of obsolescence of its customers' rivals to sell its software to the major competitors in each industry. Among the large corporations who began to adopt R/2 were Dupont, General Mills, Goodyear Tire and Rubber, Heinz, and Shell Oil, as well as 80 of the 100 largest companies in Germany, such as Hoechst, Daimler Benz, and BASF. By 1991 R/2 had gone through four releases or versions and had established itself as the standard for integrated corporate business software in Europe. By 1994, SAP could claim more than 1,400 R/2 installations worldwide.

R/3 Development in the 1980s

As R/2's potential began to peak in the mid-1980s, Plattner and company's former employer, IBM, announced a new "system applications architecture" (SAA) technology in which all IBM operating systems and platforms would be fully harmonized such that code written for one product would work with any other. Seeing the ramifications of such integratability for its own products, in 1987 SAP began developing R/3 for use in the decentralized, non-mainframe computing environment known as client-server. In client-server arrangements, data is processed not by a single costly mainframe but by many cheaper networked "server" computers, which display their data on flexibly arrangeable PCs called "clients." While R/2 focused on providing data processing solutions for static, individual functions of business operations, such as inventory tracking or shipping, R/3 was designed to allow a business to view its entire business operation as a single integrated process in which data entered into any single application in the system would simultaneously be registered in every other. In theory, a company's entire data network would now be a cohesive, interpretable whole that would enable management to more efficiently allocate resources, develop products, manage inventory, forecast trends, streamline manufacturing processes, and automate routine operations.

R/3 itself consisted of IBM's OS/2 operating system as its "front end" or user interface, IBM's DB2 program as its database component, and SAP's own proprietary application component, which was based on AT&T's Unix operating system because it offered the greatest functionality with other vendors' systems. Thus was created the three-tiered architecture--interface or desktop + database + application--on which all later versions of R/3 would be based.

Introduction of R/3 in the 1990s

After five years in development, R/3 had been launched with the expectation that it would complement R/2's multinational-oriented niche by extending SAP's reach into the mid-sized, less mainframe-dominated business software market. Unexpectedly, however, R/3's release coincided with a growing trend toward corporate downsizing, and even SAP's largest customers began eyeing R/3 as a less labor-intensive replacement for R/2. As a result, in the space of one year (1992-93), the percentage of SAP America's total revenue generated by R/3 catapulted from five to 80 percent, and R/2's status as SAP's flagship product dwindled from 95 percent of revenues to only 20 percent. R/3 was suddenly hot, and virtually overnight SAP had translated its reputation as Germany's wunderfirma to the global stage.

On the strength of R/3's rocketing sales, by the mid-1990s SAP had traveled from the relative anonymity of 1992 to the business applications vendor of choice for nine of the ten largest U.S. corporations, one-third of the Fortune 500, seven of the ten largest Business Week Global 1000, and 80 percent of the Fortune 100 companies in software, computers, peripherals, and semiconductors. Total sales revenues had nearly tripled between 1991 and 1995 to DM 2.7 billion and had increased 66 percent between 1993 and 1994 alone. Such major corporations as Apple, Chevron, Colgate-Palmolive, Digital Equipment, and Polaroid were jumping on the R/3 bandwagon, and by September 1995 SAP could claim over 1,100 installations of R/3 for companies with \$1 billion or more in sales (in addition to 700 R/2 installations) and more than 1,300 installations in smaller companies (with 800 R/2 installations). SAP's share price had in the meantime grown 1,000 percent since its introduction on the German stock exchange in 1988, and by 1996 it ranked as the highest valued company in Germany.

Competition and Other Challenges

SAP's two major competitors, Oracle Systems (United States) and Baan (the Netherlands), were meanwhile making inroads into SAP's market share. Although Oracle's database product was the program most often used as R/3's database component--making SAP the largest value-added reseller of Oracle products--in 1995 Oracle announced it would overtake SAP as the world's leading provider of industry-specific software within three years. Baan, though dwarfed by SAP in sales and customers, scored major coups in the mid-1990s when both Boeing and German giant Siemens Nixdorf rejected R/3 in favor of Baan's quick-installing business software package. SAP board member Henning Kagermann dismissed the setbacks, telling the Deutsche Presse Agentur, "If SAP wins a large order, it's accepted as natural. When we lose a potential customer, immediately it's a big headline." SAP management also dispelled the severity of the threat posed by Oracle, pointing out that in head-to-head competition SAP still won the contract 80 percent of the time.

Two public relations disasters in the mid-1990s suggested not only the extent of the controversy that had begun to surround R/3 but also SAP's saavy in handling criticism. In March 1995 the German business magazine *Wirtschaftswoche* published an article accusing SAP of accepting commissions from hardware vendors for computers sold to SAP customers and quoting several users' disparaging remarks about the expense and installation time required by R/3. As share prices nose-dived, SAP lashed back. Its hardware partners unanimously denied any kickback arrangement with SAP, and SAP itself took out a court order on the magazine for inaccuracy and deliberate misquotation and ran four-page ads in major German print

outlets in which the article's sources claimed they were misquoted and expressed satisfaction with R/3. Then, in early 1996, U.S. computer industry analyst Forrester Research published a study in which it argued that SAP's R/3 was based on an obsolete architecture that could not keep pace with the open, nonproprietary architecture increasingly favored by the software industry. SAP, Forrester claimed, knew that R/3 would be obsolete by 1997 and secretly planned to foist a brand new "object-based" system called R/10 on its customers in 1999, masking its deployment through a series of add-ons to R/3. All SAP customers, Forrester advised, should minimize their dependency on R/3 and prepare "exit strategies" to avoid being trapped into an expensive installation of a new SAP product.

SAP reacted by prematurely releasing quarterly financial figures showing that R/3 sales had not in fact peaked and by vehemently denying that it was planning to abandon R/3. It further vowed to spend DM3 billion on R&D over the next five years and announced plans for new versions of R/3 that reflected its willingness to make the product, which was based on its own proprietary programming language, more open to integration with other vendors' products. SAP, moreover, signaled it was embracing the Internet-driven trend toward "object-oriented" software in which applications could be embedded with other vendors' mini-programs (called "objects" or "applets"). The strategy worked, and Forrester Research was soon announcing that SAP was "leading in the new Internet game."

The Future

In the mid-1990s industry observers agreed that SAP's continued dominance of the client-server business software market rested on its ability to stay ahead of the breathtaking pace of change in the global software market. In the mid-1990s, for example, SAP was directly affected by the rise of the "intranet," a microcosmic version of the Internet created by companies as in-house data networks, mirroring the structure and appearance of the World Wide Web but protected from the cyber surfing public by so-called firewalls. By seeming to offer the potential to perform many of the same business applications and data processing features of R/3, such intranets represented a plausible threat to SAP's market leadership. SAP responded by announcing new features that would turn R/3 into an Internet-capable tool. Using a browser connected to the Web, for example, two companies with R/3 installed in their systems could process orders in real time over the Internet, while consumers could order products electronically from a company's online catalog and be confident the order was registered immediately in the company's R/3 system.

SAP's ability to sustain its success also depended on its willingness to continue working, À la Microsoft, with its hundreds of strategic partner firms throughout the computer and services industries. SAP's Platform Partners program, for example, had enabled it to cooperate with computer manufacturers such as Compaq and IBM in tailoring SAP products to new hardware developments. And its partnership program with such Big Six accounting firms as Arthur Andersen and Price Waterhouse had spawned a lucrative new subindustry of R/3 consultants whose institutional independence from SAP enabled it to focus more of its resources on improving its product. Finally, SAP's participation with other software vendors in industry-wide initiatives (such as the Open Application Group) to determine standards for new technologies demonstrated its willingness to cooperate with potential competitors to ensure the continued functionality and influence of its products.

Significantly, in 1994 SAP formed an alliance with America's software giant Microsoft to make SAP software integratable with such Microsoft products as Windows NT, an operating system for networked computers, and SQL Server, a database product. In 1995, Microsoft returned the favor by selecting R/3 for its global finance and accounting data system. In early 1996, Microsoft founder and chairman Bill Gates paid a symbolic visit to SAP AG's German headquarters to talk up the two megacompanies' budding relationship. "We love SAP," he said. "SAP has had more impact on our general product direction than any other software company we have worked with.... [Microsoft and SAP] are the two best companies to be in."

By learning how to quash media and public relations flare-ups and better market its products, by continuing to modify R/3 to capitalize on new technologies like the Internet, and by encouraging third-party vendors to develop specialized add-on applications to extend the number of business areas in which R/3 could be used, SAP appeared to have positioned itself to remain a formidable presence in the global business software market.

SAP R/2: is 2-tier architecture. In which all 3 layers [Presentation + Application +Database] are installed in two separate systems/server. (Server One – Presentation, Server Two – Application +Database)

SAP R/3: is 3-tier architecture. In which all 3 layers [**Presentation** + **Application** +**Database**] are installed in three separate systems/server. (Server One – Presentation, Server Two – Application, Server Three-Database)

SAP R/3 : Three-Tier Architecture

With SAP R/3, SAP users in a new generation of enterprise software — from mainframe computing (client-server architecture) to the three-tier architecture of database, application, and user interface.

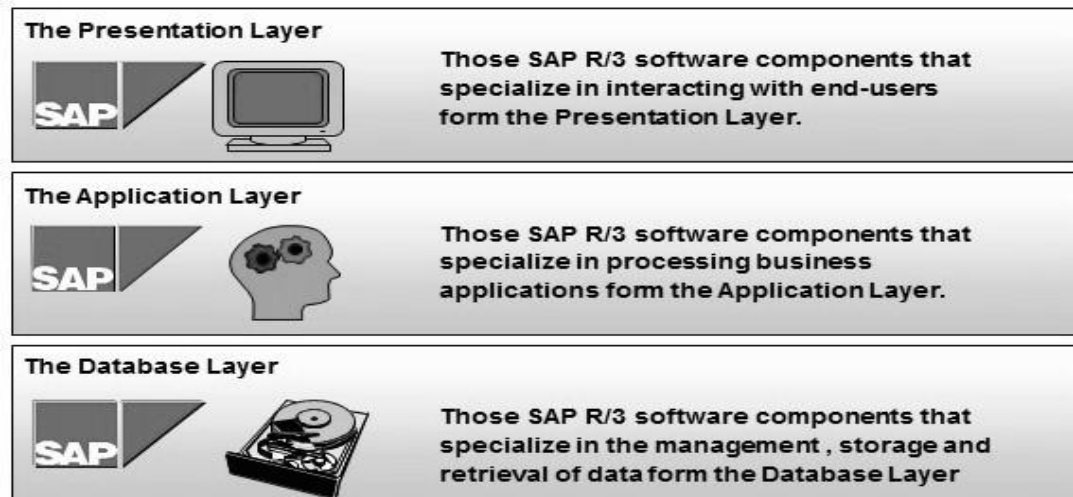
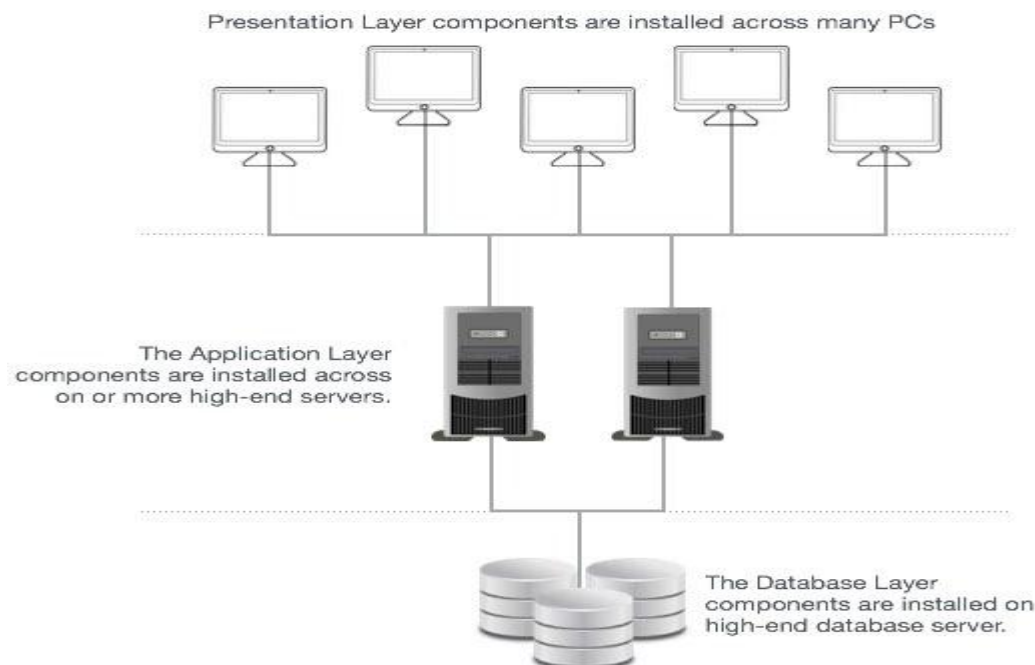


Fig: Three-Tier Architecture of SAP R/3

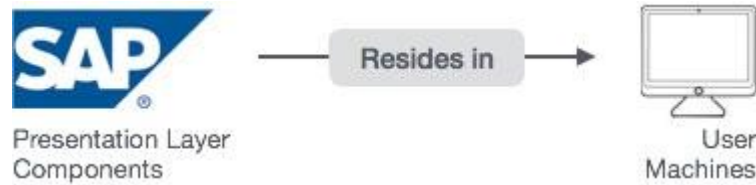


Three-Tier Architecture

Presentation Layer Servers

Presentation servers contain systems capable of providing a graphical interface.

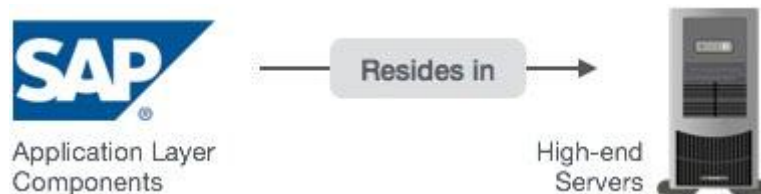
- Presentation Layer is also known as client Layer
- Presentation Layer is a user interaction
- In SAP-User interaction purpose we use GUI
- GUI stands for Graphical user interface
- Example – Desktop, Mobile Devices, laptops



Application Layer Servers

Application servers include specialized systems with multiple CPUs and a vast amount of RAM.

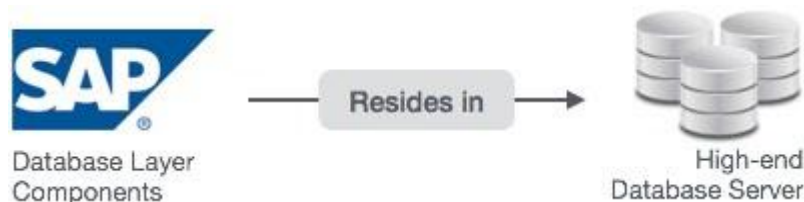
- Application Layer is also known as Kernel Layer and Basic Layer.
- SAP application programs are executed in Application Layer.
- Application Layer serves as a purpose of a communicator between Presentation and Database Layer.
- Application server is where the dispatcher distributes the work load to the different work processes makes the job done.



Database Layer Servers

Database servers contain specialized systems with fast and large hard-drives.

- Database layer stores the data
- Data store can be Business data, SAP system data, SAP tables, Programs.
- Examples – Oracle, Microsoft SQL Server, IBM DB/2, Siebel, Sybase, etc.



2. Explain in detail about SAP R/3 – CHARACTERISTICS

.SAP R/3 – CHARACTERISTICS

- SAP's R/3, introduced in 1992, is the most used ERP system in the world.
- The R/3 software package is designed to allow businesses to effectively and efficiently operate a variety of business processes within a single integrated information system.

- The software is customizable using SAP's proprietary programming language, ABAP/4. R/3 is scalable and highly suited for many types and sizes of organizations and runs on six different platforms.
- SAP's R/3 has been designed to be the best ERP system in the four areas of human resources, financial, supply chain management, and marketing. R/3 is also an international product, and meets the local fiscal, language, and tax requirements of most countries.
- SAP's R/3 is very versatile, as it will operate on six different platforms, including the recently added Microsoft NT.
- The R/3 package includes several very attractive features like it has a three-tier client/server system. Providing three tiers offers scalability and easier adaptation to the specific needs of large companies and fast-growing companies.
- SAP's R/3 is available in 14 different languages (German, English, Spanish, etc.) and also incorporates multiple currency features that provide essential information processing capabilities for multinational corporations.
- R/3's modules are organized by the functional areas of financial, human resources, supply chain management, and marketing. While information is entered separately for each specific module, the modules are fully-integrated and provide real-time applications. This means that data entered into one module is immediately and automatically updated and reflected in all of the functional areas.
- R/3 is composed of a single, virtual file structure with no subsystems.
- In addition, SAP has released "MySAP.com" which is software that provides for data interaction and processing connections with the Web.
- Financial and managerial accounting tools in SAP R/3 are contained in the financial accounting (FI) and the controlling (CO) modules. The General Ledger function in the FI module provides a comprehensive record of all information needed for external financial reporting. The accounting data is complete and accurate because the SAP system fully integrates all business transactions that were entered from all the operational areas of a company. In addition to the FI and CO modules, the SAP system includes the Investment Management (IM), Sales and Distribution (SD), Materials Management (MM), and Human Resources (HR) modules.

- Management accounting tools in SAP R/3 are cost center accounting, internal orders, product costing, and activity based costing, profitability analysis and profit center accounting.
- SAP R/3's accounting features are modeled on German approaches to accounting, and thus they are well-organized and very efficient in processing accounting information and providing accounting statements and financial reports.
- As stated previously, R/3 offers multiple currency features and a three-tier system that is capable of meeting very high demands from the accounting system for either transaction processing or financial reporting.
- SAP was the first to implement integrated treasury capabilities. This attractive feature allows a corporate treasury department to function as an in-house bank by automating the control of cash flow, investment trades, and portfolio management.
- R/3 provides check writing capability in its Accounts Receivable component which very few other programs offer.
- Additionally, there is equal access to all data in the system. This means that personnel can access financial data directly from a computer screen rather than physically meet with the treasurer, controller, or some other similar person. In other words, R/3 offers real-time, immediately updated reporting.
- R/3 also provides for a "single data entry point" where the data entered from any location is instantly sent to all other appropriate modules in the ERP system.
- The accounts payable component of SAP R/3 contains four types of transaction blocks namely:
 - ✓ The audit block
 - ✓ The receiving block
 - ✓ The vendor block
 - ✓ A manual block
- These blocks make it much less likely that improper payments will occur.
- SAP R/3 is organized with the concept that a business operates as a series of processes, which means that the company implementing R/3 may have to change and reorganize itself to properly fit with R/3 and use it effectively. Thus SAP R/3 on the whole as stated above gives:
- Significant cost and time savings.

- Minimum operating costs: no retention of redundant data in the back office.
- High level of stability and performance: response times are consistently under one second.
- Good user interface available which makes system user friendly and requires no training for the end user.

3. Explain in detail about SAP R/3 – Architecture

SAP R/3 ARCHITECTURE

Presentation layer/tier/server:

It is the interface to a user. This is the only layer from where users connect to the SAP system. DIAG (Dynamic Information Action Gateway) is the protocol which is used to communicate b/w user and SAP system. Using this we can have, our own font settings, and our own languages settings. It is user friendly. With the help of message server which identifies favorite server and logs onto it. It is intelligent server. It is operating system & db independent.

Presentation layer is nothing but SAP GUI: SAP GUI is to facilitate users to log into R/3 system. This logon can be used to all the components of SAP (CRM, APO, BW, XI etc.)

Types of SAP GUI:

-
- SAP GUI for Windows.
 - SAP GUI for HTML.
 - SAP GUI for JAVA.
-

SAP GUI for Windows: It is for the windows environment. Support platforms Includes windows 98, windows NT4, Windows 2000 and Windows XP.

SAP GUI for HTML: Front end requires only a web browser, and it is necessary to convert the presentation into HTML.

SAP GUI for JAVA: It is used only where java is supporting. It supports Windows 98, windows NT4, Windows 2000 and Windows XP, MacOS 9, MacOS x Linux, HP UX, Solaris, AIXOS/2.

Application layer/tier/server:

It is used to Provides business areas and Configure work process and Reduce traffic on DB. It is used Configure memory areas and Business logic & presentation logic handled. It consists of dispatcher, work processes, memory areas, buffer areas and interpreters.

Dispatcher: There will be only one dispatcher per instance. This is used to handle the user requests. Dispatcher receives the users request and keeps them in the queue (dispatcher queue) based on the available free resources, user request will be assigned with work process on FIFO basis. Dispatcher runs by an executable disp+work.exe. This can be monitor by using a command line tool DPMON (It listens on the port 32<sysnr>). Dispatcher assigns the user request to a dialog work process, so it will distribute request to respected work process.

Dialog process: It is used for handling generation of reports, updating the temporary tables, updating the spool tables, updating the background tables so that update, spool background processes reads those tables for execution. If the request is long running job then it will assigns to its relevant work process. Dialog work process run time is restricted to 600 sec to 1800 sec based on the parameter (rdisp/max_wprun_time).

Update work process: This process is used to update the database initially update requests are handled by dialog work process as they couldn't execute within the specified time, it is called asynchronous update process. If the task has been moved to update work process then first dialog process updates the temporary tables (VBHDR, VBDATA, VBERR, VBMOD) update process reads the temporary tables and update the database.

Enqueue process: Enqueue process is used to lock and unlock SAP objects. It will update the database and takes the users request. In order to handle this mechanism SAP has defined enqueue and dequeue (unlock) modules. Enqueue process will issue locks to message server to all the dialog instances. That is dialog communicates with message server & message server in term talks to enqueue to get the lock. Dialog process communicates with the message server and message server communicates to enqueue. Dialog processes on central instance can communicate with enqueue directly to obtain locks.

Background Process: The long running, time consuming and expensive reports or updates will be used to schedule in the non-dialog mode using the background process. Dialog work process receives the background request & updates background request & updates background job tables. Background work process reads the job tables for every 60 sec & executes them

Message Server: Message server is used to communicate with all the available dispatchers under the port number 3600+sys no. If logon load balance is configured, message server identifies the least loaded server in the logon group. It is run by an

executable msg_server.exe. This is also used to communicate with enqueue to issue locks to the work process coming from dialog instance.

Gateway: There will be one gateway work process for each instance. Gateway is used to communicate with external system.

Spool Process: Spool process is used to output the documents to the printer, fax, email, pager and sms. Dialog process receives the spool request and updates spool tables or stores spool data at OS level. Spool process reads the spool tables or spool data and output to specific device.

Note: All the work process runs with executable disp+work.exe.

Memory Areas: In order to define a work process we should have enough resources at the rate of 75mb to 150mb for each work process. When the user request is assigned to a work process, work process requires certain amount of memory to execute the user request.

Eg: Roll memory, extended memory and heap memory Buffer.

Interpreters:

1. ABAP Interpreter: This is used to interpret the ABAP code embedded in the user request.
2. Screen Interpreter: This is used to interpret the screens.
3. SQL Interpreter: This is used to interpret SQL Statements in the ABAP program.

Dispatcher: It receives user request and assigns work process or keep user request in dispatcher queue.

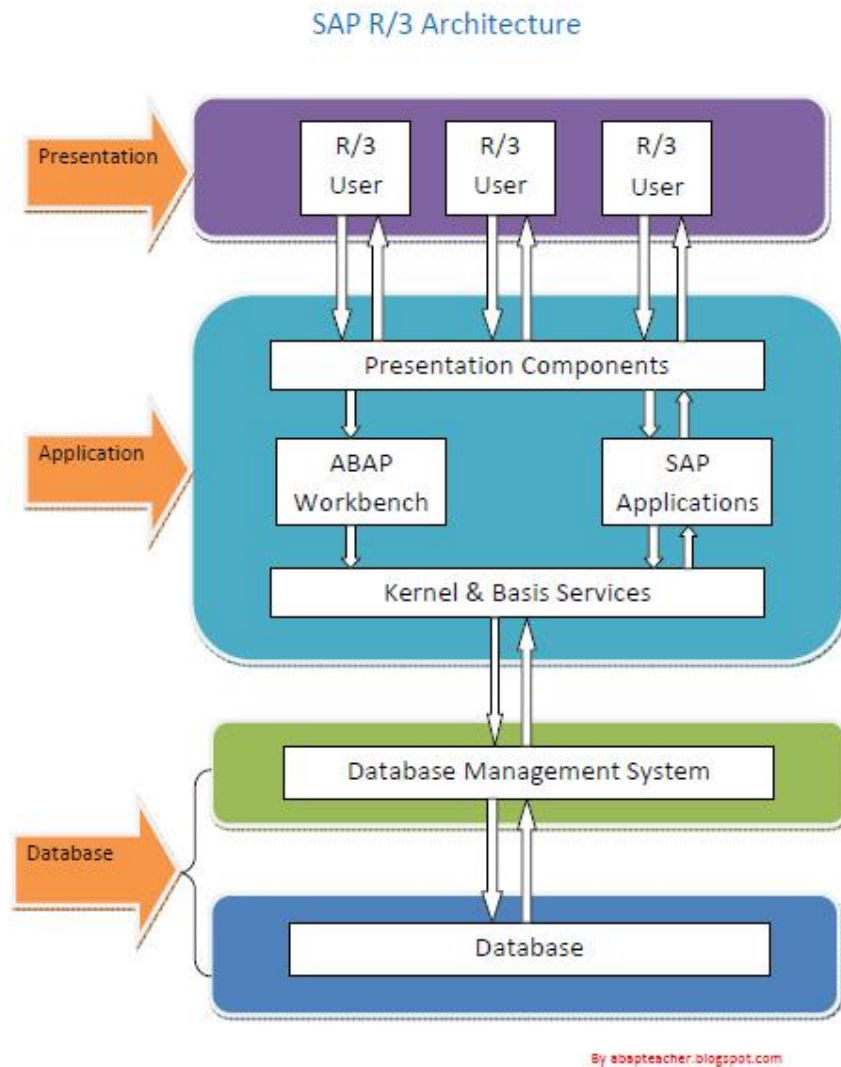
Task Handler: It is the agent which processes the user request by segregating into screen, ABAP, SQL interpreters.

User Context: The user context is the buffer area where it stores user logon attributes, authorization parameters.

Dispatcher Queue: It is the queue where user exists when work processor is busy. It follows FIFO.

Database Layer/tier:

It is the layer where database is hosted. It has its own memory areas, buffer areas, work processes etc. A central RDBMS realizes the database layer of SAP R/3 systems. Initially SAP database will use open SQL but database client will convert open SQL into native SQL. That is the reason SAP supports different databases.

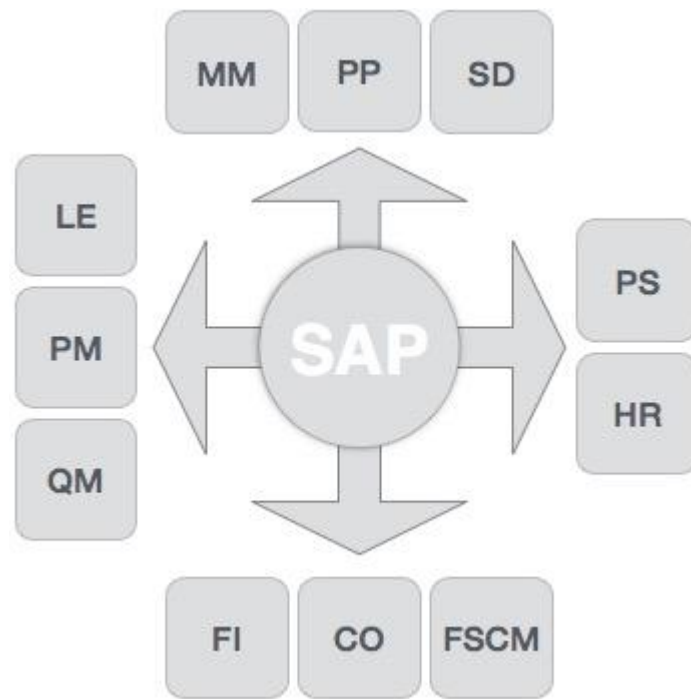


4. Explain in detail about SAP Modules.

SAP MODULES

SAP solutions include a number of functional modules, which support transactions to execute key business processes, such as

- Financial Accounting (FI)
- Financial Supply Chain Management (FSCM)
- Controlling (CO)
- Materials Management (MM)
- Sales and Distribution (SD)
- Logistics Execution (LE)
- Production Planning (PP)
- Quality Management (QM)
- Plant Maintenance (PM)
- Project System (PS)
- Human Resources (HR)



Finance and Controlling (FICO)

SAP FICO is a combination of two ERP modules, i.e., Finance Accounting (FI) and Controlling (CO). Under Finance in SAP and at an enterprise level, the following modules take part –

- FI – Finance
- CO – Controlling
- IM – Investment Management
- TR – Treasury
- EC – Enterprise Controlling

SAP FI (Financial Accounting) is accountable for tracking the flow of financial data across the organization in a controlled manner and integrating all the information for effective strategic decision-making.

Activities Involved in SAP FI

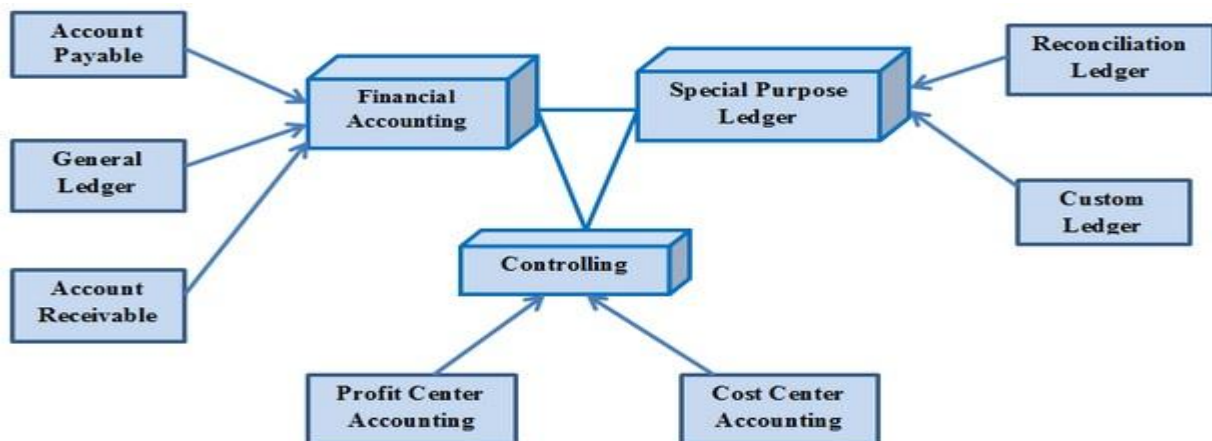
- Creation of Organizational Structure (Defining Company, Company Codes, business Areas, Functional Areas, Credit Control, Assignment of Company Codes to Credit Controls)
- Financial Accounting Global Settings (Maintenance of Fiscal Year, Posting Periods, defining Document types, posting keys, Number ranges for documents)
- General Ledger Accounting (Creation of Chart of Accounts, Account groups, defining data transfer rules, creation of General Ledger Account)
- Tax Configuration & Creation and Maintenance of House of Banks

- Account Payables (Creation of Vendor Master data and vendor-related finance attributes like account groups and payment terms)
- Account Receivables (Creation of Customer Master data and customer-related finance attributes like account groups and payment terms)
- Asset Accounting
- Integration with SD and MM

SAP CO (Controlling) module facilitates coordinating, monitoring, and optimizing all the processes in an organization. It controls the business flow in an organization. This module helps in analyzing the actual figures with the planned data and in planning business strategies.

Activities Involved in SAP CO

- Cost Element Accounting (Overview of the costs and revenues that occur in an organization)
- Cost Center Accounting
- Activity-Based-Accounting (Analyzes cross-departmental business processes)
- Internal Orders
- Product Cost Controlling (Calculates the costs that occur during the manufacture of a product or provision of a service)
- Profitability Analysis (Analyzes the profit or loss of an organization by individual market segments)
- Profit Center Accounting (Evaluates the profit or loss of individual, independent areas within an organization)



Sales & Distribution Management (SD):

SAP SD is one of the most important modules in SAP. It has a high level of integration complexity. SAP SD is used by organizations to support sales and distribution activities of products and services, starting from enquiry to order and then ending with

delivery. SAP SD can monitor a plethora of activities that take place in an organization such as products enquires, quotation (pre-sales activities), placing order, pricing, scheduling deliveries (sales activity), picking, packing, goods issue, shipment of products to customers, delivery of products and billings.

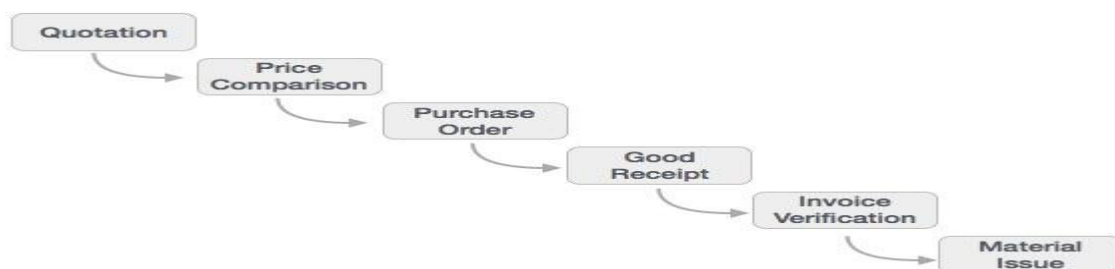
In all these processes, multiple modules are involved such as FI (Finance Accounting), CO (Controlling), MM (Material Management), PP (Production Planning), LE (Logistics Execution), etc., which shows the complexity of the integration involved.

Activities Involved in SAP SD

- Setting up Organization Structure (creation of new company, company codes, sales organization, distribution channels, divisions, business area, plants, sales area, maintaining sales offices, storage location)
- Assigning Organizational Units (Assignment of individual components created in the above activities with each other according to design like company code to company, sales organization to company code, distribution channel to sales organization, etc.)
- Defining Pricing Components (Defining condition tables, condition types, condition sequences)
- Setting up sales document types, billing types, and tax-related components
- Setting up Customer master data records and configuration

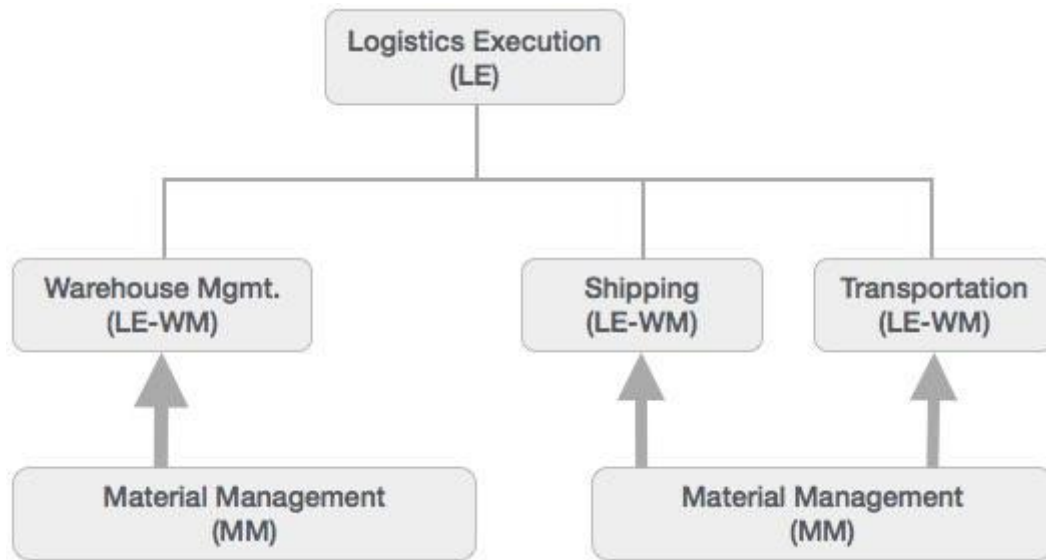
Material Management (MM)

Material Management deals with movement of materials via other modules like logistics, supply chain management, sales and delivery, warehouse management, production and planning.



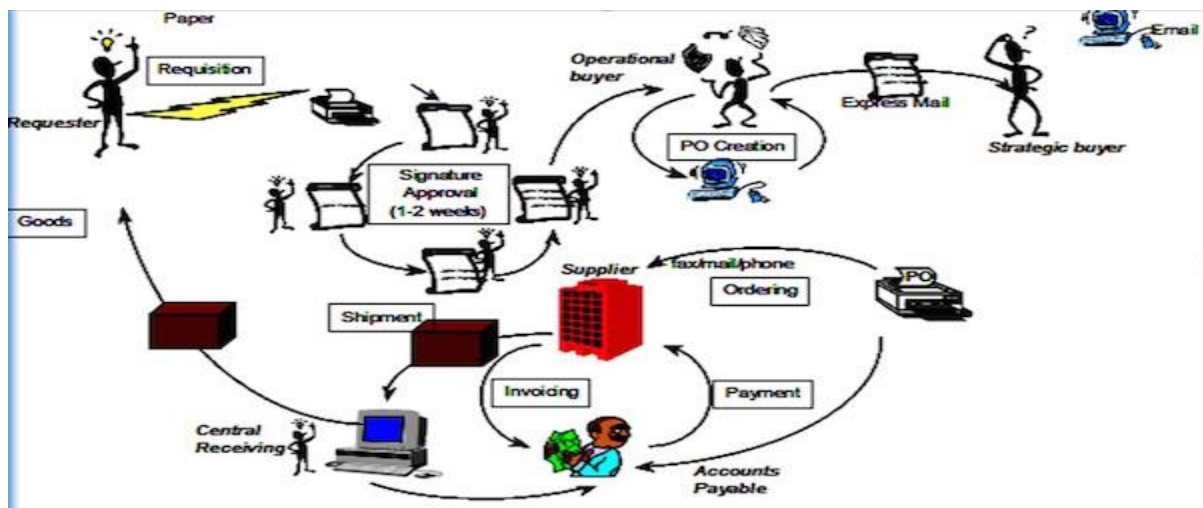
Logistic Execution (LE)

Logistic Execution can be divided into two sub-modules, i.e., shipment of goods (purchase to procurement process) and warehouse management (storage of goods). These two modules are integrated with sale and distribution, material management, and production and planning.



Supplier Relationship Management (SRM)

As the name SRM suggests, this module deals with the effective and efficient transition of products and services between an organization and its suppliers. The main process covered in this section is procurement of products like direct materials, indirect materials, and services. This module can effectively integrate with planning, accounting, and inventory system.



Customer Relationship Management (CRM)

CRM deals with end-to-end customer related processes. CRM is designed to centralize the data related to all the customers associated with an organization. It helps an organization –

- Maintain its sales, services, and build marketing strategies according the market demand and customer data analysis.
- Remain focused on its customers and via information analysis, help the business to know more about its customers.
- Improve sales and services and building better relationships with customers.



Human Resource (HR)

The most important objective of master data administration in Human Resources is to enter employee-related data for administrative, time-recording, and payroll purposes. A new employee can be hired without using Recruitment. Instead you can hire someone by running a personnel action in Personnel Administration, thereby creating the necessary data for the employee to be hired. Employee data must be kept current. After an employee is hired, circumstances can always arise which necessitate either the entry of new data or the correction of current data. For instance –

- An employee moves to his or her new address must be stored in the system.
- An employee gets a pay hike at the start of the year. The new salary must be stored for the relevant date.
- An employee changes jobs within the organization. His or her organizational assignment, working time, and salary also change.
- Data can be stored for the past, present, or future.

Note – Entering payroll-relevant data in the past triggers retroactive accounting.



The HR module is comprised of major areas of functionality known as sub-modules. The HR module is a true demonstration of the strength of the SAP product in Enterprise Resource Planning.

The HR system has very strong integration points (where data is passed back and forth without human intervention) with just about all of the other SAP modules. In addition, there is very tight integration amongst the HR sub-modules.

5. Explain in detail about SAP NetWeaver.

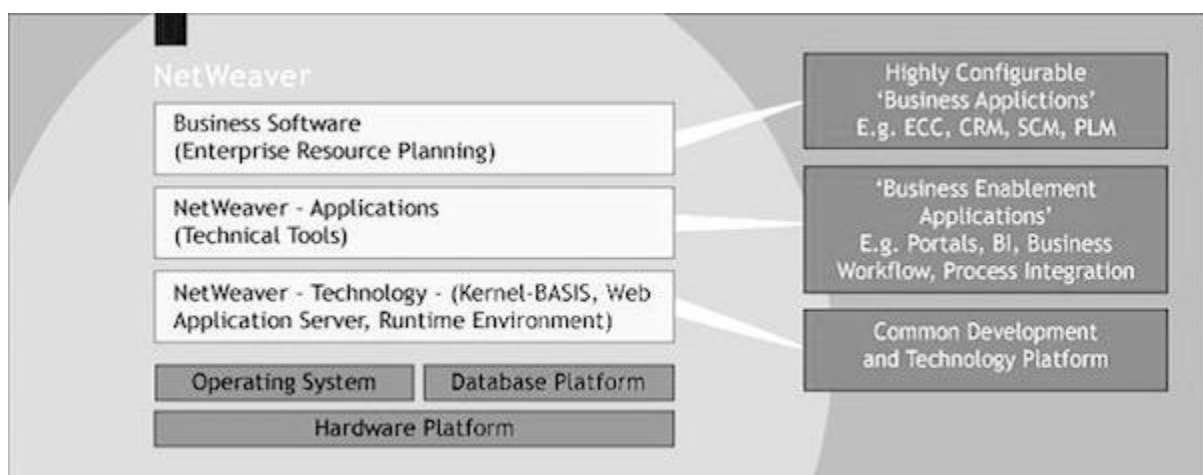
SAP - Net Weaver

NetWeaver is a combination of the underlying SAP Kernel (also known as the SAP OS layer, basically the WEB AS) and any SAP software tool for business enablement.

NetWeaver at a Glance

SAP NetWeaver describes all the software and services used for 'Business Enablement'. The SAP Business suite, such as ECC or SRM, contains the software components for that specific business solution. SAP NetWeaver is an open technology platform that offers a comprehensive set of technologies for running mission-critical business applications and integrating people, processes, and information.

SAP NetWeaver is a web-based, open integration, application platform that serves as the foundation for enterprise service-oriented architecture (enterprise SOA) and allows the integration and alignment of people, information, and business processes across business and technology boundaries. It utilizes open standards to enable integration with information and applications from almost any source or technology. SAP NetWeaver is the foundation of SAP Business Suite and SAP Business by Design. It also powers partner solutions and customer custom-built applications.



SAP NetWeaver Components

SAP NetWeaver includes a comprehensive set of components, applications, and tools.

SAP NetWeaver Application Server

It supports platform-independent web services, business applications, and standards-based development, enabling you to leverage existing technology assets for Web-services-oriented solutions.

SAP NetWeaver Business Warehouse

It enables you to integrate data from across the enterprise and transform it into practical, timely business information to drive sound decision making.

SAP NetWeaver Gateway

It enables developers to create applications that link business users to SAP software from any environment and through any device.

SAP NetWeaver Master Data Management

It ensures cross-system data consistency and helps integrate business processes across the extended value chain.

SAP NetWeaver Process Orchestration

It helps improve processes, from simple workflows to integrated processes that span applications and organizational boundaries. It includes capabilities for business process management, business rules management, and process integration.

SAP NetWeaver Portal

It unifies critical information and applications to give users role-based views that span the enterprise, enabling you to take full advantage of your information resources.

SAP Auto-ID Infrastructure

It gives you all the capabilities you need to integrate all automated sensing devices including RFID readers and printers, Bluetooth devices, embedded systems, and barcode devices.

SAP NetWeaver Identity Management

It addresses access and provisioning issues facing a typical enterprise. It creates a new opportunity for integrating business processes, and helps you to integrate systems in a heterogeneous IT environment.

SAP NetWeaver Information Lifecycle Management

It allows you to archive data in a readily accessible format according to regulatory retention rules that you define.

SAP NetWeaver Tools: SAP NetWeaver includes the following tools

Adaptive Computing Controller:

It provides a central point of control for assigning computing resources and optimizing their use.

SAP NetWeaver Composition Environment

It provides a robust environment for design, deployment, and running of composite applications that comply with a service-oriented architecture.

SAP NetWeaver Developer Studio

It offers a convenient user interface and rich functionality for developing J2EE applications.

SAP NetWeaver Visual Composer

It simplifies the creation of portal content and analytics applications, enabling business analysts to build or customize applications using a visual user interface rather than manual coding.

SAP Solution Manager

It facilitates technical support for distributed systems with functionality that covers all key aspects of solution deployment, operation, and continuous improvement.

SAP NetWeaver Applications: SAP NetWeaver includes the following applications

SAP NetWeaver Enterprise Search

It provides a simple and secure gateway to enterprise objects and transactions.

SAP NetWeaver Single Sign-On

It offers a comprehensive single sign-on solution, enabling reuse of a person's initial authentication for subsequent log-ins to all applications.

5. Explain in detail about Customer Relationship Management.

CUSTOMER RELATIONSHIP MANAGEMENT (CRM)

- Customer relationship management (CRM) is a term that refers to practices, strategies and technologies that companies use to manage and analyze customer interactions and data throughout the customer lifecycle, with the goal of improving business relationships with customers, assisting in customer retention and driving sales growth.
- CRM systems are designed to compile information on customers across different channels -- or points of contact between the customer and the company -- which could include the company's website, telephone, live chat, direct mail, marketing materials and social media.
- CRM systems can also give customer-facing staff detailed information on customers' personal information, purchase history, buying preferences and concerns.
- Customer relationship management (CRM) entails all aspects of interaction that a company has with its customers, whether it is sales or service-related.

- While the phrase customer relationship management is most commonly used to describe a business-customer relationship (B2C), CRM systems are also used to manage business to business to business (B2B) relationships. Information tracked in a CRM system includes contacts, clients, contract wins and sales leads and more.

CRM software

CRM software consolidates customer information and documents into a single CRM database so business users can more easily access and manage it. The other main functions of this software include recording various customer interactions (over email, phone calls, social media or other channels, depending on system capabilities), automating various workflow processes such as tasks, calendars and alerts, and giving managers the ability to track performance and productivity based on information logged within the system.

Common features of CRM software include:

- **Marketing automation:** CRM tools with marketing automation capabilities can automate repetitive tasks to enhance marketing efforts to customers at different points in the lifecycle. For example, as sales prospects come into the system, the system might automatically send them marketing materials, typically via email or social media, with the goal of turning a sales lead into a full-fledged customer.
- **Sales force automation:** Also known as sales force management, sales force automation is meant to prevent duplicate efforts between a salesperson and a customer. A CRM system can help achieve this by automatically tracking all contact and follow-ups between both sides.
- **Contact center automation:** Designed to reduce tedious aspects of a contact center agent's job, contact center automation might include pre-recorded audio that assists in customer problem-solving and information dissemination. Various software tools that integrate with the agent's desktop tools can handle customer requests in order to cut down the time of calls and simplify customer service processes.
- **Geolocation technology or location-based services:** Some CRM systems include technology that can create geographic marketing campaigns based on customers' physical locations, sometimes integrating with popular location-based GPS apps. Geolocation technology can also be used as a networking or contact management tool in order to find sales prospects based on location.

CRM Strategy

Customer relationship management is often thought of as a business strategy that enables businesses to improve in a number of areas. The CRM strategy allows you to do the following:

- Understand the customer
- Retain customers through better customer experience
- Attract new customers
- Win new clients and contracts
- Increase profitability
- Decrease customer management costs

The Benefits of CRM

- The biggest benefit most businesses realize when moving to a CRM system comes directly from having all your business data stored and accessed from a single location.
- Before CRM systems, customer data was spread out over office productivity suite documents, email systems, mobile phone data and even paper note cards and Rolodex entries.
- Storing all the data from all departments (e.g., sales, marketing, customer service and HR) in a central location gives management and employees immediate access to the most recent data when they need it.
- Departments can collaborate with ease, and CRM systems help organization to develop efficient automated processes to improve business processes.
- Other benefits include a 360-degree view of all customer information, knowledge of what customers and the general market want, and integration with your existing applications to consolidate all business information.

6. Explain in detail about SAP Business Information Warehouse

Business Information Warehouse (Business Warehouse or SAP BW)

Business Warehouse (BW) is a combination of databases and database management tools that are used to support management decision making.

The reporting, analysis, and interpretation of business data is of central importance to a company in guaranteeing its competitive edge, optimizing processes, and enabling it to react quickly and in line with the market.

As a core component of SAP NetWeaver, the **SAP Business Warehouse** provides data warehousing functionality, a business intelligence platform, and a suite of business intelligence tools that enable businesses to attain these goals. Relevant business information from productive SAP applications and all external data sources can be integrated, transformed, and consolidated in BW with the toolset provided.

Business Warehouse provides flexible reporting and analysis tools to support you in evaluating and interpreting data, as well as facilitating its distribution. Managers are able to make well-founded decisions and determine target-orientated activities on the basis of this analysis.

Advantages:

On one hand SAP Business Information Warehouse's comprehensive Business Content enables a quick and cost-effective implementation.

On the other hand, it provides a model that can be used as a guideline during implementation using experience gained from other implementations.

- Business Information Warehouse (sometimes shortened to "Business Warehouse" or BW) is a packaged, comprehensive business intelligence product centered around a data warehouse that is optimized for (but not limited to) the R/3 environment from SAP
- SAP Business Warehouse (also known as SAP NetWeaver Business Warehouse or SAP BW) is the cornerstone of SAP's strategic Enterprise Data Warehouse solutions and runs on industry standard RDBMS and SAP's HANA in-memory DBMS.
- Like most data warehouses, BW is a combination of databases and database management tools that are used to support management decision making.
- BW supplies the infrastructure typical of data warehouses, but also includes preconfigured data extractors, analysis and report tools, and business process models.
- Among the other features of BW are: Business Application Programming Interfaces (BAPIs) that enable connections to non-R/3 applications; preconfigured business content; an integrated OLAP processor; automated data extraction and loading routines; a metadata repository; administrative tools; multiple language support; and Business Explorer, a Web-based user interface.

- SAP Business Warehouse is an integral component of the company's mySAP Business Intelligence group of products.
-

7. Explain in detail about SAP Advanced Planning and Optimization

SAP Advanced Planning and Optimization (SAP APO)

- ✓ APO (Advanced Planning and Optimization) application is at the heart of SCM.
- ✓ It offers planning and optimization functionalities in different business processes of Demand Planning, Supply Planning, Supply and Demand Matching, Production Planning Detailed Scheduling, Global Available to Promise and Transportation Management.
- ✓ SAP APO functionalities enable Plan to Inventory End to End Business Processes.
- ✓ Collaboration with Suppliers and Customers is also possible through newer application Supply Network Collaboration (SNC) till recently known as Inventory Collation Hub (ICH)
- ✓ With SCM 5.0 a new set of functionalities under Services Parts Planning were added specifically catering to Spare Parts Management.
- ✓ APO as a application is tightly integrated to execution (OLTP) system like ERP using a standard interface called Core Interface Function (Interfaces - Core Interface (CIF) and BAPIs). It also has full BI (erstwhile BW) component for Data Mart as well as Reporting functionalities.

SAP Advanced Planning and Optimization (SAP APO) provides a fully integrated range of functions that you require for planning and executing your logistic processes. SAP APO supports the following:

- ✓ Intercompany interaction on a strategic, tactical, and operative planning level
- ✓ Collaboration with logistic partners from order receipt through stock monitoring to product shipping
- ✓ Maintenance of relationships with customers and business partners
- ✓ Continuous optimization and measurement of the performance of the logistic network

Purpose

- ✓ SAP Advanced Planning and Optimization (SAP APO) offers a fully integrated pallet of functions that you use to plan and execute your supply chain processes. SAP APO supports the following:
- ✓ Business collaboration on a strategic, tactical, and operational planning level
- ✓ Co-operation between partners at all stages of the supply chain process; from order receipt, stock monitoring, through final shipping of the product
- ✓ Cultivation of customer and business partner relationships
- ✓ Constant optimization and evaluation of the supply chain network's efficiency

Features

SAP Advanced Planning and Optimization (SAP APO) can be deployed as part of an SAP SCM Server installation or as an add-on to SAP ERP (as of SAP enhancement package 6 for SAP ERP 6.0).

An SAP SCM Server installation provides the complete scope of SAP APO components and functions. In an add-on deployment to SAP ERP, the following SAP APO components and functions are available:

- ✓ Demand Planning (SCM-APO-FCS)
- ✓ Supply Network Planning (SCM-APO-SNP)
- ✓ Production Planning and Detailed Scheduling (SCM-APO-PPS)
- ✓ Global Available-to-Promise (SCM-APO-ATP)

8. Explain in detail about ABAP & its Workbench Tools**ABAP (ADVANCED BUSINESS APPLICATION PROGRAMMING)**

ABAP – the programming language used in SAP for developing business application support and development. ABAP is a programming language that runs in the SAP ABAP runtime environment, created and used by SAP for the development of application programs including:

- ✓ Reports
- ✓ Module Pool Programming
- ✓ Interfaces
- ✓ Forms
- ✓ Data conversions
- ✓ User Exits & BADI

All of R/3's applications and even parts of its basis system were developed in ABAP. ABAP is an event-driven programming language. User actions and system events

control the execution of an application. ABAP is also called ABAP/4. The “4” in ABAP/4 stands for “Fourth Generation Language” or 4GL.

ABAP Workbench

The ABAP Workbench is used by SAP for the development of standard and custom application software. The ABAP Workbench is also used to create dictionary objects. It consists of the following components –

- ✓ **ABAP Editor** is used to maintain programs.
- ✓ **ABAP Dictionary** is used to maintain Dictionary objects.
- ✓ **Repository Browser** is used to display a hierarchical structure of the components in a package.
- ✓ **Menu Painter** is used to develop graphical user interfaces including menu bars and toolbars.
- ✓ **Screen Painter** is used to maintain screen components for online programs.
- ✓ **Repository Information System** contains information about development and runtime objects, such as data models, dictionary types and table structures, programs, and functions.
- ✓ **Test and Analysis Tools**, such as the Syntax Check and the Debugger.
- ✓ **Function Builder**, which allows to create and maintain function groups and function modules.
- ✓ **Data Modeler**, a tool which supports graphical modeling.
- ✓ **Workbench Organizer**, which maintains multiple development projects and manages their distribution.



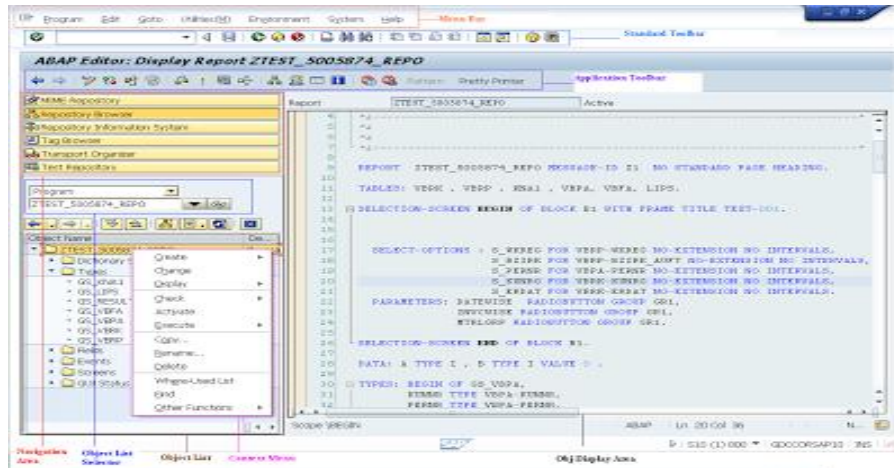
ABAP WORKBENCH TOOLS

The ABAP Workbench is a collection of tools used to develop, test and run ABAP programs. It is a Programming environment GUI in SAP to develop different business applications using ABAP language.

ABAP Workbench Tools - Quick Guide

Object Navigator

Object Navigator is the central point of entry into ABAP workbench as you can access any object of SAP system through it. In SAP all the Development objects are properly arranged together in an object list under some category such as packages, global class, programs etc. The transaction code to Open Object navigator is **SE80**.



Components of Object Navigator Interface

We can choose browsers from Object Navigation area list. The various browsers are:-

1. **MIME Repository**- Multipurpose Internet Mail Exchange Files Repository- It displays all the directories with the MIME files which were imported into current system.
2. **Repository Browser** - It displays repository objects (Please note that all ABAP programs are Objects to SAP) in the form of object list which are organized by selection categories like programs, packages, classes etc. This is the default displayed browser by object navigator.
3. **Repository Information System**- Unlike Repository browser it displays all the available objects from information system without any search category.
4. **Tag Browser** displays all the TAGs for web-apps.
5. **Transport Organizer**- It displays the Transport request sent to it by user based on the request or task number.
6. **Test Repository**- Displays results of the test cases after testing repository objects.

Object Navigator facilitates users to perform the following tasks:

- ✓ Select a browser and navigate in the object list.
- ✓ Use the tools for development objects .
- ✓ Navigate from one window to another window.
- ✓ Perform syntax checks in the integrated window.
- ✓ Open an object in a new session using an Additional dialog box.

ABAP Editor

SAP has provided Transaction SE38 for ABAP editor. All the reports and includes and other programs are created/edited using this transaction. ABAP pers spend most of their time in the ABAP editor.

✓ **Function Builder**

Transaction SE37 is used for accessing function modules. Through this we can access all the SAP standard function modules and we can also create our own 'Z' function modules.

✓ **Class Builder**

Similarly to the function Builder is the class builder which is used for ABAP objects programming. We can access all the SAP standard classes and can as well create our classes using class builder. Transaction code SE24 is used for this purpose.

✓ **Screen Painter**

Screen Painter is mainly used for either creating forms or Dialog Programs. Screen Painter provides us with various tools like Text I/P O/P, box, table wizards etc which could be used to create our own GUI screens for programs. Screen painter is accessible via transaction code SE51.

✓ **Menu Painter**

Menu Painter is used to design the user interfaces for the programs. Using Menu Painter we can customize the user menu. By default the SAP provides the user with all the available options for all the 'Z' programs we create. Now to change this or limit the user options we create our pf_status using transaction SE41 (Menu Painter) and set the same PF_STATUS in our program.

✓ **Message Maintenance**

Messages helps SAP system to communicate with the user. Messages of types Information-I, Error-E, Warning-W, Success-S could be displayed using message classes. To create a message class could be done at transaction- SE91. Then we could use the message class in our reports/Programs.

ABAP Dictionary

ABAP Dictionary is one of the important tools of ABAP Workbench. It is used to create and manage data definitions without redundancies. ABAP Dictionary always provides the updated information of an object to all the system components. The presence and role of ABAP Dictionary ensures that data stored in an SAP system is consistent and secure. ABAP dictionary could be accessed via transaction SE11.

9. Explain in detail about ABAP Data Dictionary

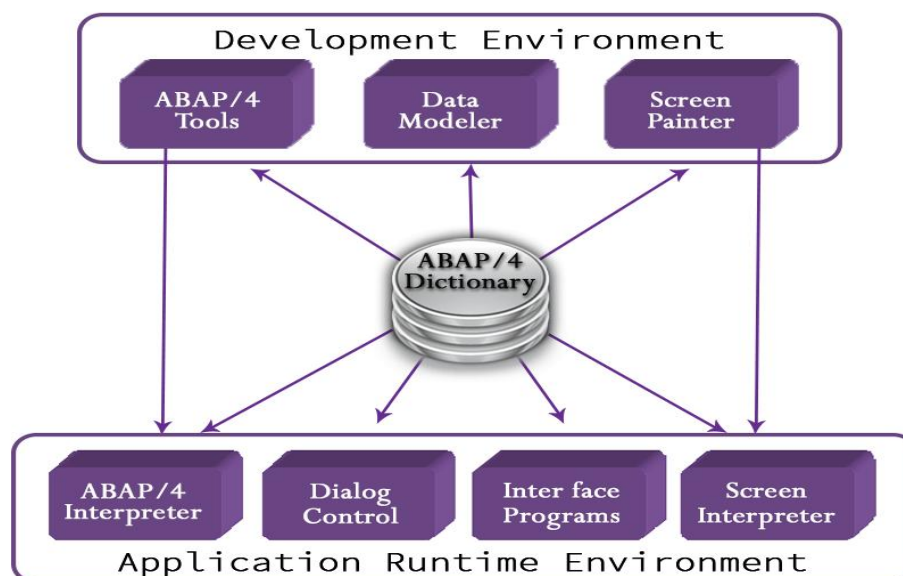
ABAP Data Dictionary

A data dictionary is a central source of information for the data in a information management system. Its main function is to support the creation and management of data definitions (or "metadata").

Data dictionary is used for

- ✓ Management of data definitions
- ✓ Provision of information for evaluations
- ✓ Support for software development
- ✓ Support for documentation
- ✓ Ensuring that data definitions are flexible and up-to-date

INTEGRATION INTO ABAP/4 WORKBENCH



Objects in the ABAP Dictionary resided on three levels that support their re-usability.

These levels are:

- ✓ Tables and structures
- ✓ Data elements

- ✓ Domains

Domains

- ✓ Describes the technical characteristics of a table field
- ✓ Specifies a value range which describes allowed data values for the fields
- ✓ Fields referring to the same domain (via the data elements assigned to them) are changed when a change is made to the domain
- ✓ Ensures consistency
- ✓ Ex. Purchasing document number (EBELN)

Data Elements

- ✓ Describes the role played by a field in a technical context
- ✓ Fields of same semantic meaning can refer to the same data element
- ✓ Contains the field information

Tables

- ✓ Represent the Database Tables where data actually resides.
- ✓ Tables can be defined independently of the database in the ABAP Dictionary.
- ✓ The fields of the table are defined with their (database-independent) SAP ABAP data types and lengths.

Structures

- ✓ Are record declarations that do NOT correspond to a Database Table.
- ✓ Just like user-defined data type.
- ✓ Defined like a table and can then be addressed from ABAP programs.
- ✓ Structures contain data only during the runtime of a program.

Aggregated Objects of ABAP Dictionary

Aggregated means consisting of several components. In the ABAP Dictionary, aggregated objects are objects which come from several different transparent tables.

- ✓ Views
- ✓ Search Help
- ✓ Lock Objects

Views

- ✓ Views in SAP _ ABAP are used to summarize data which is distributed among several tables
- ✓ The data of a view is not actually physically stored. The data of a view is instead derived from one or more other tables
- ✓ It is tailored to the needs of a specific application

Search Help

- ✓ A Search help is a tool to help you search for data records in the system

- ✓ An efficient and user-friendly search assists users where the key of a record is unknown

Lock Objects

- ✓ Simultaneous accessing of the same data record by two users in the SAP system is synchronized by a lock mechanism.
- ✓ Locks are set and released by calling certain function modules. These function modules are generated automatically from the definition of so-called lock objects in the ABAP/4 Dictionary
- ✓ **Function modules** :Enqueue_<obj name> - to lock the table dequeue_<obj name> - to release the lock

10. Explain in detail about ABAP Repository Information System

ABAP/4 REPOSITORY INFORMATION SYSTEM

The description of the ABAP/4 Repository Information System (in short: Repository Information System) given here is restricted to aspects of relevance for work with the Data Modeler. Using the ABAP/4 Repository Information System, you can search for all modeling objects of the Data Modeler that you wish to display or edit.

Repository Information System: overview

- ✓ The ABAP/4 Repository Information System provides you with the two basic functions Find and Where-used list.
- ✓ The Find function allows you to find objects of a specific object class that correspond to certain selection criteria. You can, for example, search for a list of all entity types belonging to a particular development class.
- ✓ The Where-used list function allows you to determine the other objects in which a particular object is used. For example, you could search for all data models in which a specific entity type occurs.

Repository Information System: access

- ✓ From the initial screen of the Data Modeler, there are two ways of calling the ABAP/4 Repository Information System.

Call the Repository Information System from the Data Modeler

1. from the menu

- ✓ You can call the ABAP/4 Repository Information System from the initial screen of the Data Modeler by selecting **EnvironmentRepository Info Sys**. The screen listing the areas of the ABAP/4 Repository Information System is displayed. Most of the areas are preceded by a symbol. This indicates that subareas exist. To display these subareas, select the line you are interested in followed by **EditExpand sub tree**. The subareas hidden beneath the nodes are displayed.
- ✓ To allow searching for modeling objects, position the cursor on **Modeling** and select **Edit Expand sub tree**. Under the heading **Data modeling**, you will find all the points relating to the Data Modeler, namely **Data models**, **Entity types**, and **Entity type attributes**. When you double-click on one of these points, the corresponding selection screen appears.

2. from a modeling object

- ✓ Leaving the entry field **Modeling object** empty, select the object class you require under **Selection** followed by the menu option **Find**.
- ✓ The corresponding selection screen (for entity types or data models) appears.
- ✓ You can call the ABAP/4 Repository Information System from other points in the Data Modeler. You will be informed of this at the appropriate points of this documentation.

REPOSITORY INFORMATION SYSTEM: SEARCHING FOR OBJECTS

- ✓ You can search for objects in the Data Modeler with the Repository Information System.

Search for objects with the Repository Information System

- ✓ Call the relevant selection screen in the ABAP/4 Repository Information System.
- ✓ The standard selections for the object class in question are displayed.
- ✓ The maximum number of hits to be selected is also shown. These values are preset in your user settings.
- ✓ If you wish to display all the objects from a particular object class, enter * in the first search field on the first line. Then select the menu options **Program Execute**.

- ✓ A list corresponding to your selection will be output.
- ✓ If you wish to search via fields that are not included in the standard selections, you can display all available selections for a particular object class.
- ✓ To do so, choose **all selections**. The additional selection options are now displayed underneath the standard selections.
- ✓ To search for objects with particular attributes, you have to make the relevant entries in the search fields provided. The way in which these entries are analyzed is determined via the arrow pushbutton next to the input field.
- ✓ You can search for single values or for ranges of values. If you want to search for an object with a specific attribute, it is sufficient to enter this attribute.
 1. If you wish to search for all data models with names starting with U, it is sufficient to enter **U*** in the field **Data model**. The option **Pattern** is then selected automatically.
 2. If you wish to search for all data models whose names start with a letter before U in the alphabet and after X, on the line **Data model** you must enter **U** in the first field and press the arrow pushbutton. A dialog box appears. In the second field, enter **X**. Position the cursor on each field and click on **Options**. Select the relevant search criteria and start the search.

Repository Information System: settings

- ✓ The selection options to be defaulted when you are searching for objects and the maximum number of objects matching the search criteria to be displayed are laid down in your user settings

Define the user settings in the Repository Information System

Defining the user settings

- ✓ To define your user settings, you have to call the initial screen of the ABAP/4 Repository Information System.
- ✓ To do so, call the ABAP/4 Development Workbench from the R/3 initial screen with **Tools ABAP/4 Workbench**.
- ✓ In the ABAP/4 Development Workbench select **Overview Repository Info Sys**.

Setting a start variant

- ✓ The selection options available to you are determined by the choice of variant. A standard variant is normally set. You can change the variant by selecting **Settings User parameters** in the initial screen of the ABAP/4 Repository Information System.

- ✓ A dialog box containing a number of different variants appears. Select the variant you require and click on the **Save** pushbutton. The selected variant is now adopted as your start variant.

Specifying the maximum number of hits

- ✓ The maximum number of hits determines the maximum number of objects matching the search criteria that is to be selected. For example, if 100 is entered as the maximum number of hits and you are searching for all tables with names beginning with U, only the first 100 tables matching this selection criterion will be selected. You can change the standard entry for the maximum number of hits.
- ✓ To do so, select **Settings** **User parameters** in the initial screen of the ABAP/4 Repository Information System. A dialog box appears in which you can enter the value you require. Click on **Save**. Your selection for the maximum number of hits is now adopted as your standard setting.

Repository Information System: selection options

- ✓ By specifying selection options for a field, you can determine how the entry you have made is analyzed during the search process.

Set the selection options in the Repository Information System

- ✓ Press the arrow pushbutton beside the relevant field. A dialog box appears. Click on the **Options** pushbutton to maintain the selection options:
- ✓ **Single value**
- ✓ All entries matching the entry value are selected.
- ✓ **Greater than or equal**
- ✓ All entries greater than or equal to the entry value are selected.
- ✓ **Less than or equal**
- ✓ All entries with values less than or equal to the entry value are selected.
- ✓ **Pattern**
- ✓ All entries matching the pattern are selected. This option is displayed only when you have made a generic entry in the field, such as U* in the field **Data model**.
- ✓ **Exclude pattern**
- ✓ All entries that do not match the pattern are selected. This option is displayed only if you have made a generic entry in the field, such as U* in the field **Data model**.

- ✓ Not equal
- ✓ All entries not equal to the entry are selected.
- ✓ Less than
- ✓ All entries with values that are less than the entry value are selected.
- ✓ Greater than
- ✓ All entries with values greater than the entry value are selected.

Repository Information System: examples for selection options

- ✓ When selecting data models, **U** is entered in the field **Data model**. Then the arrow pushbutton is used and the **Options** pushbutton chosen.
- ✓ After selecting **ProgramExecute**, the output list that is displayed varies according to the selection options set for the field **Data model**.

Single value

- ✓ No table is selected, since there is no table of the name U.
- ✓ **Greater than or equal**: All tables with names starting with U or with a letter occurring after U in the alphabet are selected.
- ✓ **Less than or equal**: All tables with names starting with U or with a letter occurring before U in the alphabet are selected.
- ✓ **Not equal**: All tables with names that do not begin with U are selected.
- ✓ **Less than**: All tables with names starting with a letter that occurs before U in the alphabet are selected.
- ✓ **Greater than**: All tables with names starting with a letter occurring after U in the alphabet are selected.
- ✓ When selecting data models, **U*** is entered in the field **Data model**. The selection options **Pattern** and **Exclude pattern** result in the following selections being made:
 - ✓ **Pattern**
 - ✓ All tables with names starting with U are selected.
 - ✓ **Exclude pattern**
 - ✓ All tables with names that do not begin with U are selected.

Repository Information System: where-used list

This function allows you to determine the other objects in which a particular object is used.

Use the Repository Information System to find out where objects are used

- ✓ Access the ABAP/4 Repository Information System.

- ✓ To do so, select **ToolsABAP/4 Workbench** in the initial screen of the R/3 System. In the ABAP/4 Development Workbench, select **Overview Repository Info Sys**.
- ✓ To find out where modeling objects are used, position the cursor on **Modeling** and select the menu options **Edit Expand sub tree**.
- ✓ Under the heading **Data modeling**, you will find all the points relating to the Data Modeler, namely **Data models**, **Entity types**, and **Entity type attributes**.
- ✓ Click on **Data models** or **Entity types** and select **Repository Infosys Where-used list**.
- ✓ A dialog box appears in which you have to enter the name of the object for which you wish to see a where-used list. In the case of entity types, you also have to specify which type of where-used list you require (i.e. for use in data models or in tables).
- ✓ Once you have done so, click on **Cont**. A hit list is displayed.
- ✓ You can execute the following functions from within this list.
- ✓ You can access the display screen for an object by selecting the object you are interested in choosing **Display**.
- ✓ You can access the maintenance screen for an object from within the list by selecting the object in the first column of the display and selecting **Change**.
- ✓ You can display the occurrences of an object from the list in other objects by selecting the object and choosing **Utilities Where-used list**.

11. Explain in detail about Structure of ABAP Programs

STRUCTURE OF ABAP PROGRAMS:

ABAP processing logic is responsible for processing data in R/3 application programs. ABAP was designed specifically for dialog-oriented database applications. The following sections deal with how an ABAP program is structured and executed. ABAP programs are responsible for data processing within the individual **dialog steps** of an application program. This means that the program cannot be constructed as a single sequential unit, but must be divided into sections that can be assigned to the individual dialog steps. To meet this requirement, ABAP programs have a modular structure. Each module is called a **processing block**. A processing block consists of a set of ABAP statements. When you run a program, you effectively call a series of processing blocks. They cannot be nested.

The following diagram shows the structure of an ABAP program:



Each ABAP program consists of the following two parts:

Declaration Part for Global Data, Classes and Selection Screens

- ✓ The first part of an ABAP program is the declaration part for global data, classes, and selection screens. This consists of:
- ✓ All declaration statements for global data. Global data is visible in all internal processing blocks. You define it using declarative statements that appear before the first processing block, in dialog modules, or in event blocks. You cannot declare local data in dialog modules or event blocks.
- ✓ All selection screen definitions.
- ✓ All local class definitions (CLASS DEFINITION statement). Local classes are part of ABAP Objects, the object-oriented extension of ABAP.
- ✓ Declaration statements which occur in procedures (methods, subroutines, function modules) form the declaration part for local data in those processing blocks. This data is only visible within the procedure in which it is declared.

Container for Processing Blocks

- ✓ The second part of an ABAP program contains all of the processing blocks for the program. The following types of processing blocks are allowed:
- ✓ Dialog modules (no local data area)
- ✓ Event blocks (no local data area)
- ✓ Procedures (methods, subroutines and function modules with their own local data area).
- ✓ Whereas dialog modules and procedures are enclosed in the ABAP keywords which define them, event blocks are introduced with event keywords and concluded implicitly by the beginning of the next processing block.
- ✓ All ABAP statements (except declarative statements in the declaration part of the program) are part of a processing block. Non-declarative ABAP statements,

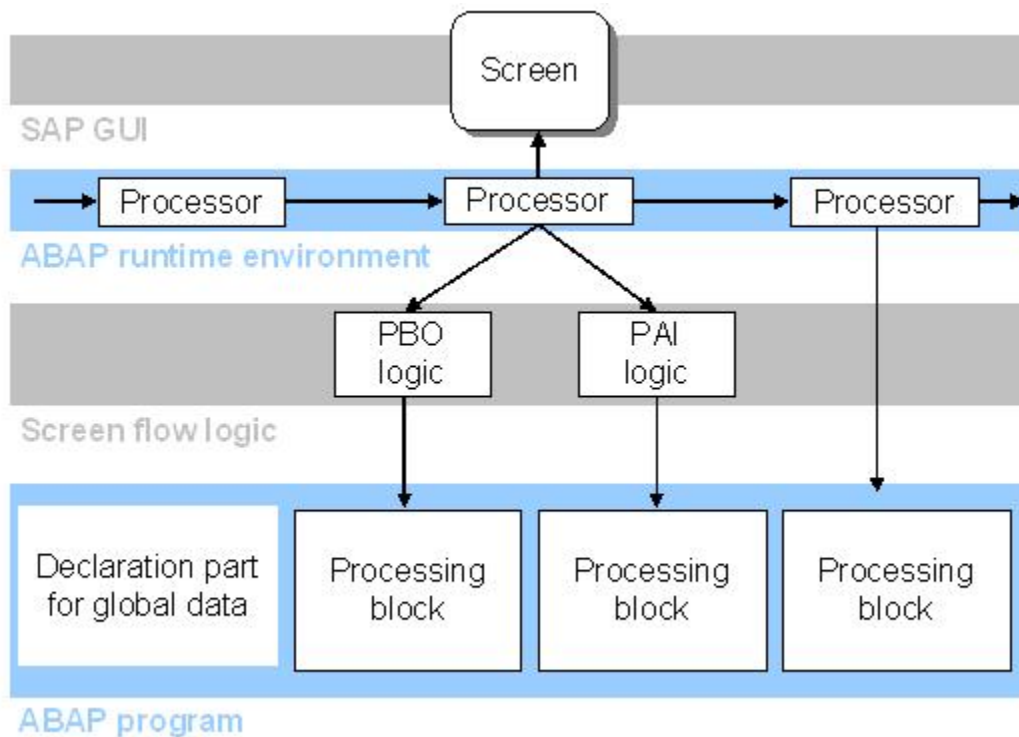
which occur between the declaration of global data and a processing block are automatically assigned to the START-OF-SELECTION processing block.

Calling Processing Blocks

- ✓ You can call processing blocks either from outside the ABAP program or using ABAP commands which are themselves part of a processing block. Dialog modules and event blocks are called from outside the ABAP program. Procedures are called using ABAP statements in ABAP programs.
- ✓ Calling event blocks is different from calling other processing blocks for the following reasons:
- ✓ An event block call is triggered by an event. User actions on selection screens and lists, and the runtime environment trigger events that can be processed in ABAP programs. You only have to define event blocks for the events to which you want the program to react (whereas a subroutine call, for example, must have a corresponding subroutine).

Program Types and Execution

When you run an ABAP program, you call its processing blocks. ABAP programs are controlled from outside the program itself by the processors in the current work process. For the purposes of program flow, we can summarize the screen processor and ABAP processor into the ABAP runtime environment. The runtime environment controls screens and ABAP processing blocks. It contains a range of special control patterns that call screens and processing blocks in certain orders. These sections are also called processors. When you run an ABAP program, the control passes between various processors.



In the R/3 System, there are various types of ABAP program. The program type determines the basic technical attributes of the program, and you must set it when you create it. The main difference between the different program types is the way in which the runtime environment calls its processing blocks.

When you run an application program, you must call at least the first processing block from outside the program, that is, from the runtime environment. This processing block can then either call further processing blocks or return control to the runtime environment. When you start an ABAP program, the runtime environment starts a processor (dependent on the program type), which calls the first ABAP processing block. An ABAP program can be started either by the user or by the system (for example, in background processing), or through an external interface (for example, Remote Function Call).

There are two ways of allowing users to execute programs - either by entering the program name or by entering a transaction code. You can assign a transaction code to any program. Users can then start that program by entering the code in the command field. Transaction codes are also usually linked to a menu path within the R/3 System.

The following program types are relevant to application programming:

Type 1: Type 1 programs have the important characteristic that they do not have to be controlled using user-defined screens. Instead, they are controlled by the runtime environment, which calls a series of processing blocks (and selection screens and lists

where necessary) in a fixed sequence. User actions on screens can then trigger further processing blocks.

You can start a type 1 program and the corresponding processor in the runtime environment using the SUBMIT statement in another ABAP program. There are also various ways of starting a type 1 program by entering its program name. This is why we refer to type 1 programs as executable programs.

When you run a type 1 program, a series of processors run in a particular order in the runtime environment. The process flow allows the user to enter selection parameters on a selection screen. The data is then selected from the database and processed. Finally, an output list is displayed. At no stage does the programmer have to define his or her own screens. The runtime environment also allows you to work with a logical database. A logical database is a special ABAP program which combines the contents of certain database tables. The flow of a type 1 program is oriented towards reporting, whose main tasks are to read data from the database, process it, and display the results. This is why executable programs (type 1) in the R/3 System are often referred to as **reports**, and why running an executable program is often called **reporting**.

Since it is not compulsory to define event blocks, you can yourself determine the events to which your ABAP program should react. Furthermore, you can call your own screens or processing blocks at any time, leaving the prescribed program flow. You can use this, for example, to present data in a table on a dialog screen instead of in a list. The simplest executable program (report) contains only one processing block (START-OF-SELECTION).

Executable programs do not require any user dialog. You can fill the selection screen using a variant and output data directly to the spool system instead of to a list. This makes executable programs (reports) the means of background processing in the R/3 System.

You can also assign a transaction code to an executable program. Users can then start it using the transaction code and not the program name. The reporting-oriented runtime environment is also called when you run a report using a transaction code. This kind of transaction is called a **report transaction**.

It is appropriate to use executable programs (reports) when the flow of your program corresponds either wholly or in part to the pre-defined flow of the runtime environment. Until Release 4.5A, the only way to use a logical database was to use an executable program. However, from Release 4.5A, it is also possible to call logical databases on their own.

Type M: The most important technical attribute of a type M program is that it can only be controlled using screen flow logic. You must start them using a transaction code, which is linked to the program and one of its screens (initial screen). Another feature of these programs is that you must define your own screens in the Screen Painter (although the initial screen can be a selection screen).

When you start a program using a transaction code, the runtime environment starts a processor that calls the initial screen. This then calls a dialog module in the corresponding ABAP program. The remainder of the program flow can take any form. For example, the dialog module can:

- ✓ return control to the screen, after which, the processing passes to a subsequent screen. Each screen has a following screen, set either statically or dynamically.
- ✓ call other sequences of screens, selection screens or lists, from which further processing blocks in the ABAP program are started.
- ✓ call other processing blocks itself, either internally or externally.
- ✓ call other application programs using CALL TRANSACTION (type M program) or SUBMIT (type 1 program).

ABAP programs with type M contain the dialog modules belonging to the various screens. They are therefore known as **module pools**. It is appropriate to use module pools when you write dialog-oriented programs using a large number of screens whose flow logic largely determines the program flow.

Type F: Type F programs are containers for **function modules**, and cannot be started using a transaction code or by entering their name directly. Function modules are special procedures that you can call from other ABAP programs.

Type F programs are known as **function groups**. Function modules may only be programmed in function groups. The **Function Builder** is a tool in the ABAP Workbench that you can use to create function groups and function modules. Apart from function modules, function groups can contain global data declarations and subroutines. These are visible to all function modules in the group. They can also contain event blocks for screens in function modules.

Type K: You cannot start type K programs using a transaction code or by entering the program name. They are containers for **global classes** in ABAP Objects. Type K programs are known as **class definitions**. The **Class Builder** is a tool in the ABAP Workbench that you can use to create class definitions.

Type J: You cannot start type J programs using a transaction code or by entering the program name. They are containers for **global interface** in ABAP Objects . Type J programs are known as **interface definitions**. Like class definitions, you create interface definitions in the **Class Builder**.

Type S: You cannot start a type S program using a transaction code or by entering the program name. Instead, they are containers for subroutines, which you can call externally from other ABAP programs. Type S programs are known as **subroutine pools**. They cannot contain screens.

Type I: Type I programs - called **includes** - are a means of dividing up program code into smaller, more manageable units. You can insert the coding of an include program at any point in another ABAP program using the INCLUDE statement. There is no technical relationship between include programs and processing blocks. Includes are more suitable for logical programming units, such as data declarations, or sets of similar processing blocks. The ABAP Workbench has a mechanism for automatically dividing up module pools and function groups into include programs.

12. Explain in detail about ABAP4 Syntax:

ABAP SYNTAX

The syntax of the ABAP programming language consists of the following elements:

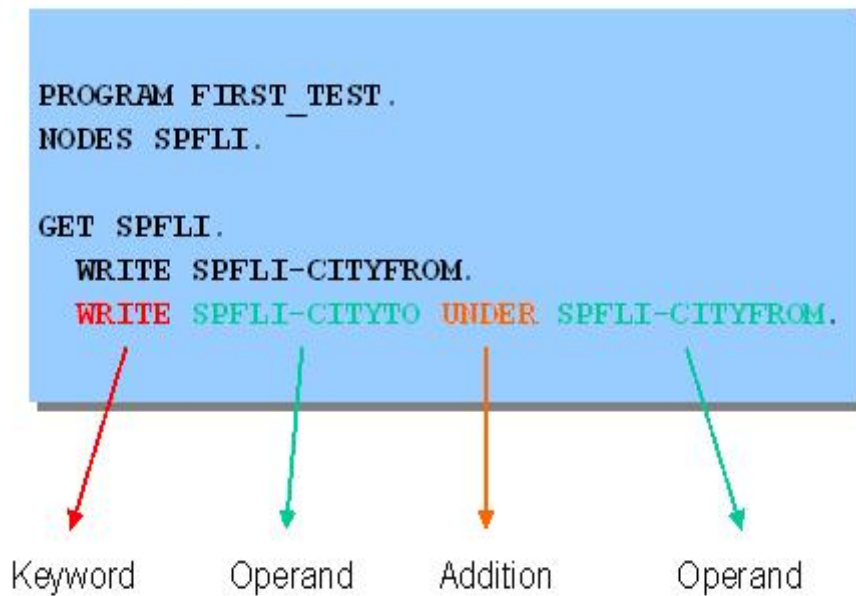
Statements

An ABAP program consists of individual ABAP statements. Each statement begins with a keyword and ends with a period.

```
PROGRAM FIRST_PROGRAM.
```

```
WRITE 'My First Program'.
```

This example contains two statements, one on each line. The keywords are PROGRAM and WRITE. The program displays a list on the screen. In this case, the list consists of the line "My First Program". The keyword determines the category of the statement. For an overview of the different categories, refer to ABAP Statements.



This diagram shows the structure of an ABAP statement.

13. Explain in detail about ABAP4 Data Types

ABAP Data Types

Data Type describes the technical characteristics of a Variable of that type. Data type is just the blue print of a variable.

Predefined ABAP Types

| DATA TYPE | DESCRIPTION | DEFAULT LENGTH | DEFAULT VALUE |
|-----------|--------------|----------------|----------------------|
| C | Character | 1 | ' ' |
| F | Numeric | 15 | ' ' |
| D | Date | 8 | '00000000' |
| T | Time | 6 | '000000' |
| X | Hexa Decimal | 16 | '0000000000000000' |
| I | Integer | 10 | '0000000000' |
| P | Packed | 18 | '000000000000000000' |

| | | | |
|--|-------|--|--|
| | float | | |
|--|-------|--|--|

User defined data types

Use **TYPES** keyword to define the data types.

```
TYPES: name(10) TYPE c,
length TYPE p DECIMALS 2,
counter TYPE i,
id(5) TYPE n.
```

Structured data types

Structured data type is grouping of several simple data types under one name. Use the keywords **BEGIN OF** and **END OF** to create a structured data type.

```
TYPES: BEGIN OF student,
id(5) TYPE n,
name(10) TYPE c,
dob TYPE d,
place(10) TYPE c,
END OF student.
```

Constants

Constants are used to store a value under a name. We must specify the value when we declare a constant and the value cannot be changed later in the program. Use **CONSTANTS** keyword to declare a constant.

```
CONSTANTS: pi TYPE p DECIMALS 2 VALUE '3.14',
yes TYPE c VALUE 'X'.
```

Existing Data Types

- ✓ **ACCP**: Posting period. The length is set to 6 places for this data type. The format is YYYYMM. In input and output, the system inserts a point between the year and month, so the template of this data type has the format '____.____'.
- ✓ **CHAR**: Character string. Fields of type CHAR can have a maximum length of 1333 in tables. If you want to use longer character fields in tables, you must choose data type LCHR. There are no restrictions on the length of such fields in structures.
- ✓ **CLNT**: Client. Client fields always have three places.

- [illegible]

overflow can also occur when writing values into the database. In this case the system throws an ABAP-OO exception.

- ✓ **DF16_RAW:** Normalized decimal floating point number. Representation based on type RAW. The values can be sorted and compared according to their numerical value, and they can be used in indexes. Database arithmetic is not available. The system supports up to 16 decimal digits in the coefficient. Decimal floating point numbers of this type are represented internally with 16 decimal places according to the IEEE-754r standard. Valid values are numbers between 1E-383 and 9.999999999999999E+384, plus the corresponding negative numbers plus zero.
- ✓ **DF16_SCL:** Scaled decimal floating point number. The difference between this type and DF16_RAW is that DF16_SCL has an additional column of type INT2 for the scale. This column is visible, but the value is written and read automatically. The values having this data type can be sorted and compared according to their numerical value, and they can be used in indexes. Database arithmetic is not available. The system supports up to 16 decimal digits in the coefficient. Decimal floating point numbers of this type are represented internally with 16 decimal places according to the IEEE-754r standard. Valid values are numbers between 1E-383 and 9.999999999999999E+384, plus the corresponding negative numbers plus zero.
- ✓ **DF16_DEC:** Decimal floating point number. Representation on the database with type DEC, length and number of decimal places must be specified by the programmer. The values have at most 15 digits on the database, with at most 14 decimal places. The advantage of this type is that database arithmetic is available. The disadvantage is that values are silently rounded to the specified number of decimal places when they are written into the database. An overflow can also occur when writing values into the database. In this case, the system throws an ABAP-OO exception.
- ✓ **INT1:** 1-byte integer between 0 and 255. The length is set to 3 places for this data type.
- ✓ **INT2:** 2-byte integer between -32767 and 32767. Fields of this type must be used only for length fields. The system positions these length fields immediately in front of a long field (type LCHR, LRAW). With INSERT or UPDATE on the long field, the database interface enters the length which was actually used in the length field. The length is set to 5 places for this data type.

- ✓ **INT4**: 4-byte integer between -2147483648 and 2147483647. The length for this data type is limited to 10 places.
- ✓ **LANG**: Language key. It has its own field format for special functions. This data type always has length 1. The language key is displayed at the user interface with 2 places, but is stored with 1 place in the database. The conversion exit ISOLA converts the display at the user interface for the database and vice versa. This conversion exit is automatically allocated to a domain with data type LANG at activation.
- ✓ **LCHR**: Character string of any length, but has to be declared with a minimum of 256 characters. You must locate fields of this type at the end of transparent tables (in each table there can be only one such field) and must be preceded by a length field of type INT2. If there is an INSERT or UPDATE in ABAP programs, this length field must be filled with the length actually required. If the length field is not filled correctly, this leads to a data loss in the LCHR field. Fields of this type cannot be used in the WHERE condition of a SELECT statement.
- ✓ **LRAW**: Uninterpreted byte string of any length, but has to be declared with a minimum length of 256. You must locate fields of this type at the end of transparent tables (in each table there can be only one such field) and must be preceded by a length field of type INT2. If there is an INSERT or UPDATE in ABAP programs, this length field must be filled with the length actually required. If the length field is not filled correctly, this leads to a data loss in the LRAW field. A field of this type cannot be used in the WHERE condition of a SELECT statement.
- ✓ **NUMC**: Long character field in which only numbers can be entered. The length of this field is limited to a maximum of 255 places.
- ✓ **PREC**: Obsolete data type. The length is set to 2 places for this data type but internally it is treated like INT2. Dynpro fields of type PREC are restricted to 2 places and must not contain a sign.
- ✓ **QUAN**: Quantity. Equivalent to an amount field DEC. A field of this type must always refer to a units field with UNIT format (reference field). The maximum length for this data type is 31 places.
- ✓ **RAW**: Uninterpreted byte string. Fields of type RAW may have only a maximum length of 255 in tables. If longer raw fields are required in tables, you should select data type LRAW.

- ✓ **RAWSTRING:** Uninterpreted byte string of variable length. In the Dictionary a length can be specified for this type (at least 256 characters). This data type can be used in types (data elements, structures, table types) and domains. You can store binary data of type RAWSTRING in the database. There are restrictions; for a description of them, refer to the documentation of the ABAP statement 'STRING'. In ABAP, this type is implemented as a reference to a storage area of variable size. The system proposes 132 characters as the default for the output length. You cannot attach search helps to components of this type.
- ✓ **STRING:** Character string with variable length This data type can be used only in types (data elements, structures, table types) and domains. In the dictionary a length can be specified for this type (at least 256 characters). It can be used in database tables only with restrictions. For a description of them, refer to the documentation of the ABAP statement 'STRING'. In ABAP, this type is implemented as a reference to a storage area of variable size. The system proposes 132 characters as default for the output length. You cannot attach search helps to components of this type.
- ✓ **SSTRING:** Short character string with variable length. In the Dictionary the number of characters can be specified for this type (from 1 to 1333). This data type can be used only in types (data elements, structures, table types) and domains. It can be used in database tables. To do so, refer to the documentation of the ABAP statement 'STRING'. In ABAP, this type is implemented as a reference to a storage area of variable size. String fields of this type can be used in indexes and in the WHERE condition of a SELECT statement. You cannot use them in table keys.
- ✓ **TIMS:** Time. The length is set to 6 places for this data type. The format is HHMMSS. The template for input and output has the form '__.__.__.__'.
- ✓ **UNIT:** Unit. Fields of this type are referenced by fields of type QUAN. The length of this data type is set to 2 or 3 places.
- ✓ **VARC:** Character field of variable length. Creation of new fields of this data type is no longer supported.

Reference Types

Reference types describe single fields that can contain references to **global** classes and interfaces from the ABAP class library.

In an ABAP program, you can use the **TYPE** addition to refer directly to a data element. The predefined Dictionary data types of the domain are then converted into the corresponding ABAP types.

14. Explain in detail about ABAP4 Tables:

Table Basics

Internal table is a data object in ABAP that exists only at run time of a program. It means when the program execution is complete then the internal table will be lost. We use internal table to store database table data after fetching it by a select query. The ABAP runtime system dynamically manages the internal table's memory. It means we developer do not need to work on memory management of internal table.

Internal table has three parts – **rows, columns & work area**.

- ✓ Rows are the line type of internal table. It is a structure which contains several fields. Those fields are of data elements. We need to declare the structure locally or globally to declare the internal table.
- ✓ Columns are the fields of internal table. Those fields are of different data elements declared by locally or globally.
- ✓ The most important part of an internal table is its work area. Work area is basically the line type of an internal table. It means it has the same structure of the rows of internal table. Work area contains the same fields of same type of the rows. It is of two types – implicit & explicit work area.
- ✓ When we declare an internal table with header line then a work area is automatically created with the same name of the table. This work area is called implicit work area which is actually the header line. There is no need to declare work area separately. This work area / header line contains the same table as of the internal table.

15. Explain in detail about Various ABAP4 Statements

1.MOVE

To assign the value of a data object **source** to a variable **destination**, use the following statement:

MOVE source TO destination.

or the equivalent statement

destination = source.

The content of **source** remains unchanged, **source** does not therefore have to be a variable - it can also be a literal, a text symbol, or a constant. You must always specify decimal points with a period (.), regardless of the user's personal settings.

Multiple assignments

f4 = f3 = f2 = f1.

are also possible. ABAP processes them from right to left as follows:

MOVE f1 TO f2.

MOVE f2 TO f3.

MOVE f3 TO f4.

In the **MOVE** statement (or when you assign one value to another with the equal sign), it is not possible to specify the field names dynamically as the contents of other fields. If you need to do this, you must use field symbols .

The source and target fields can be of different data types. The result of the value assignment depends on whether these data types are compatible and whether a type conversion can be performed. If there is no conversion rule between the data types in question, no assignment can be made.



**DATA:t(10) TYPE c,
number TYPE p DECIMALS 2,
count TYPE i.**

t = 1111.

MOVE '5.75' TO number.

count = number.

Following these assignments, the fields **t**, **number** and **count** have the values '1111 ', 5.75, and 6 respectively. When you assign the number literal 1111 to T, it is converted into a character field with length 10. When you assign **number** to **count** , the decimal number is rounded to an integer (as long as the program attribute Fixed pt. arithmetic has been set).

Assigning Values Between Components of Structures

The rules for value assignments between data objects also apply to structures. With the command

DATA: struct1 TYPE structure,

struct2 TYPE structure.

struct1 = struct2.

two structures of the same type can be assigned to one another without difficulty. Here, the entire source structure is seen as a unit and copied to the source structure. It is then possible to access the components individually again. If the structures in question are not compatible, see the conversion rules for structures.

In practice, however, you will often only need to assign certain components of a structure to be certain components of another structure. ABAP has a special statement for this purpose:

2.MOVE-CORRESPONDING sourcestruct TO destinationstruct.

This statement assigns the contents of the components of structure **sourcestruct** to the components of the **destinationstruct** structure that have **identical names**.

When it is executed, it is broken down into a set of **MOVE** statements, one for each pair of fields with identical names, as follows:

MOVE sourcestruct-comp1 TO destinationstruct-comp1.

MOVE sourcestruct-comp2 TO destinationstruct-comp2.

...

Any necessary type conversions are performed individually.



DATA: BEGIN OF address,

firstname(20) TYPE c VALUE 'Fred',

surname(20) TYPE c VALUE 'Flintstone',

initials(4) TYPE c VALUE 'FF',

street(20) TYPE c VALUE 'Cave Avenue',

number TYPE i VALUE '11',

postcode(5) TYPE n VALUE '98765',

city(20) TYPE c VALUE 'Bedrock',

END OF address.

DATA: BEGIN OF name,

surname(20) TYPE c,

firstname(20) TYPE c,

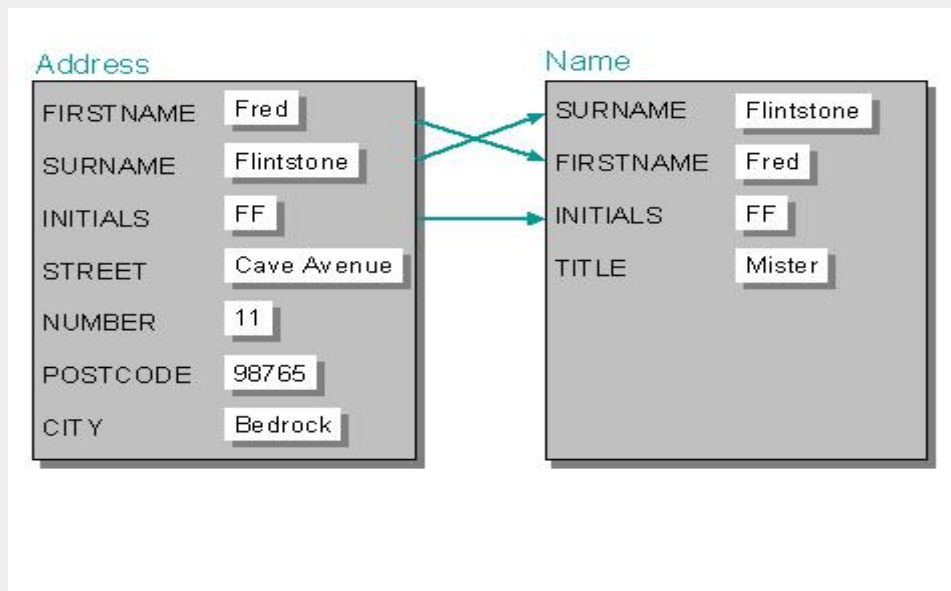
initials(4) TYPE c,

title(10) TYPE c VALUE 'Mister',

END OF name.

MOVE-CORRESPONDING address TO name.

In this example, the values of **name-surname**, **name-firstname** and **name-initials** are set to 'Flintstone', 'Fred', and 'FF'. **name-title** always has the value 'Mister'.



3. Loop structures

In a loop, a statement block is executed several times in succession. There are four kinds of loops in ABAP:

- ✓ Unconditional loops using the **DO** statement.
- ✓ Conditional loops using the **WHILE** statement.
- ✓ Loops through internal tables and extract datasets using the **LOOP** statement.
- ✓ Loops through datasets from database tables using the **SELECT** statement.
- ✓ This section deals with **DO** and **WHILE** loops. **SELECT** is an Open SQL statement, and is described in the Open SQL section. The **LOOP** statement is described in the sections on internal tables and extract datasets.

Unconditional Loops

To process a statement block several times unconditionally, use the following control structure:

DO [n TIMES] ...

[statement_block]

ENDDO.

Use the **TIMES** addition to restrict the number of loop passes to **n**.

If you do not specify any additions, the statement block is repeated until it reaches a termination statement such as **EXIT** or **STOP** (see below). The system field **sy-index** contains the number of loop passes, including the current loop pass.

You can nest **DO** loops and combine them with other loop forms.



Simple example of a **DO** loop:

DO.

WRITE sy-index.

IF sy-index = 3.

EXIT.

ENDIF.

ENDDO.

The list output is:

1 2 3

The loop is processed three times. Here, the processing passes through the loop three times and then leaves it after the **EXIT** statement.



Example of two nested loops with the **TIMES** addition:

DO 2 TIMES.

WRITE sy-index.

SKIP.

DO 3 TIMES.

WRITE sy-index.

ENDDO.

SKIP.

ENDDO.

The list output is:

1

1 2 3

2

1 2 3

The outer loop is processed twice. Each time the outer loop is processed, the inner loop is processed three times. Note that the system field **sy-index** contains the number of loop passes for each loop individually.

Conditional Loops

To repeat a statement block for as long as a certain condition is true, use the following control structure:

WHILE log_exp

[statement_block]

ENDWHILE.

log_exp can be any logical expression. The statement block between **WHILE** and **ENDWHILE** is repeated as long as the condition is true or until a termination statement such as **EXIT** or **STOP** occurs. The system field **sy-index** contains the number of loop passes, including the current loop pass.

You can nest **WHILE** loops to any depth, and combine them with other loop forms.



REPORT demo_flow_control_while.

```
DATA: length TYPE i VALUE 0,  
      str1 TYPE i VALUE 0,  
      string(30) TYPE c VALUE 'Test String'.
```

```
str1 = strlen( string ).
```

```
WHILE string NE space.
```

```
  WRITE string(1).
```

```
  length = sy-index.
```

```
  SHIFT string.
```

```
ENDWHILE.
```

```
WRITE: / 'STRLEN: ',str1.
```

```
WRITE: / 'Length of string:', length.
```

The output appears as follows:

```
T e s t   S t r i n g
```

```
STRLEN:
```

11

```
Length of String: 11
```

Here, a **WHILE** loop is used to determine the length of a character string. This is done by shifting the string one position to the left each time the loop is processed until it contains only blanks. This example has been chosen to demonstrate the **WHILE** statement. Of course, you can determine the length of the string far more easily and efficiently using the **strlen** function.

Terminating Loops

4. ABAP contains termination statements that allow you to terminate a loop prematurely. There are two categories of termination statement - those that only apply to the loop, and those that apply to the entire processing block in which the loop occurs. The **STOP** and **REJECT** statements belong to the latter group (see Exiting Eventblocks).

The termination statements that apply only to the loop in which they occur are **CONTINUE**, **CHECK** and **EXIT**. You can only use the **CONTINUE** statement in a

loop. **CHECK** and **EXIT**, on the other hand, are context-sensitive. Within a loop, they only apply to the execution of the loop itself. Outside of a loop, they terminate the entire processing block in which they occur (subroutine, dialog module, event block, and so on).

CONTINUE, **CHECK** and **EXIT** can be used in all four loop types in ABAP (**DO**, **WHILE**, **LOOP** and **SELECT**).

Terminating a Loop Pass Unconditionally

To terminate a single loop pass immediately and unconditionally, use the **CONTINUE** statement in the statement block of the loop.

CONTINUE.

After the statement, the system ignores any remaining statements in the current statement block, and starts the next loop pass.



DO 4 TIMES.

IF sy-index = 2.

CONTINUE.

ENDIF.

WRITE sy-index.

ENDDO.

The list output is:

1 3 4

The second loop pass is terminated without the **WRITE** statement being processed.

Terminating a Loop Pass Conditionally

To terminate a single loop pass conditionally, use the **CHECK condition** statement in the statement block of the loop.

5.CHECK condition.

If the condition is not true, any remaining statements in the current statement block after the **CHECK** statement are ignored, and the next loop pass starts. **condition** can be any logical expression.



DO 4 TIMES.

CHECK sy-index BETWEEN 2 and 3.

WRITE sy-index.

ENDDO.

The list output is:

2 3

The first and fourth loop passes are terminated without the **WRITE** statement being processed, because **sy-index** is not between 2 and 3.

Exiting a Loop

To terminate an entire loop immediately and unconditionally, use the **EXIT** statement in the statement block of the loop.

6.EXIT.

After this statement, the loop is terminated, and processing resumes after the closing statement of the loop structure (**ENDDO**, **ENDWHILE**, **ENDLOOP**, **ENDSELECT**). In nested loops, only the current loop is terminated.



DO 4 TIMES.

IF sy-index = 3.

EXIT.

ENDIF.

WRITE sy-index.

ENDDO.

The list output is:

1 2

In the third loop pass, the loop is terminated before the **WRITE** statement is processed.

MOVE-CORRESPONDING

MOVE-CORRESPONDING is a keyword used in SAP ABAP programming

Basic form

MOVE-CORRESPONDING rec1 TO rec2.

Effect Interprets rec1 and rec2 as field strings. If, for example, rec1 and rec2 are tables, executes the statement for their header lines.

Searches for the sub-fields which occur both in rec1 and rec2 and then generates, for all relevant field pairs which correspond to the sub-fields in ,statements of the form
MOVE rec1-ni TO rec2-ni.

The other fields remain unchanged. With complex structures, the full names of the corresponding field pairs must be identical.

Example

DATA: BEGIN OF INT_TABLE OCCURS 10,

WORD(10),
 NUMBER TYPE I,
 INDEX LIKE SY-INDEX,
 END OF INT_TABLE,
 BEGIN OF RECORD,
 NAME(10) VALUE 'not WORD',
 NUMBER TYPE I,
 INDEX(20),
 END OF RECORD.

MOVE-CORRESPONDING INT_TABLE TO RECORD.

This MOVE-CORRESPONDING statement is equivalent to both the following statements:

MOVE INT_TABLE-NUMBER TO RECORD-NUMBER.

MOVE INT_TABLE-INDEX TO RECORD-INDEX.

CHECK (ABAP Keyword)

Basic form

CHECK logexp.

Effect

CHECK evaluates

the subsequent logical expression . If it is true, the processing continues with the next statement. In loop structures like

DO ...

ENDDO

WHILE ... ENDWHILE

LOOP ... ENDLOOP

SELECT ...

ENDSELECT

CHECK with a negative outcome terminates the current loop pass and goes back to the beginning of the loop to start the next pass, if there is one.

In structures like

FORM ... ENDFORM

FUNCTION ...ENDFUNCTION

MODULE ... ENDMODULE

AT

CHECK with a negative

Outcome terminates the routine or modularization unit.

If CHECK is not in a loop or a routine or a modularization unit, a negative logical expression

terminates the current event. In contrast, the statement REJECT terminates the current event, even from loops or subroutines.

CHECK – special for reports with logical databases

Variants

1. CHECK sel.

2. CHECK

SELECT-OPTIONS.

Variant 1

CHECK sel.

Effect

Checks the

selection criterion requested by the statement SELECT-OPTIONS sel ...
.

This statement is equivalent to f IN sel , if sel was defined by SELECT-OPTIONS sel FOR f and can be used anywhere in logical expressions

If the result of this check is negative, the processing in this event is terminated and the GET events for any subordinate database tables are not processed either.

This variant of the CHECK statement should be used only if the logical database for the corresponding table does not support dynamic selections (see CHECK SELECT-OPTIONS), or SELECT-OPTIONS with the addition NO DATABASE SELECTION . Otherwise, the relevant record is not read from the database and made available to the program.

Variant 2

CHECK

SELECT-OPTIONS.

Effect

Called only after a GET event.

This statement checks all the selections for SELECT-OPTIONS where the reference field after FOR belongs to the current table dbtab (specified after GET . However,

this applies only if the logical database for dbtab does not support dynamic selections . Otherwise, the selections are passed directly to the logical database (with the exception: addition " NO DATABASE SELECTION " to SELECT-OPTIONS).

This variant of the CHECK statement only makes sense if the logical database does not support dynamic selections for the corresponding table or SELECT-OPTIONS are defined with the addition " NO DATABASE SELECTION".

You can determine from the ABAP/4 Development Workbench whether dynamic selections are defined and, if so, for which logical database tables by selecting Development -> Programming environ. -> Logical databases followed by Extras -> Dynamic selections .

Example

The logical database F1S of the demo flight reservation system contains the tables SPFLI

with, and the table SFLIGHT without, dynamic selections.

TABLES:

SPFLI,

SFLIGHT.

SELECT-OPTIONS:

SF_PRICE FOR SFLIGHT-PRICE, SP_CARR FOR SPFLI-CARRID, SP_FROM FOR SPFLI-CITYFROM NODATABASE SELECTION, SP_DEPT FOR SPFLI-DEPTIME.

Since dynamic selections are defined with the table SPFLI, but not with the table SFLIGHT, the following procedure applies:

...

GET SFLIGHT.

CHECK SELECT-OPTIONS.

This

CHECK statement is equivalent to the following statement:

CHECK

SF_PRICE.

With

GET SPFLI.

CHECK

SELECT-OPTIONS.

The CHECK statement is equivalent to the following statement:

CHECK SP_FROM.

Note

With CHECK

SELECT-OPTIONS, fields from superior tables in the database hierarchy are not (!) checked.

Note

Runtime errors

CHECK_SELOPT_ILLEGAL_OPTION:

Wrong " OPTION " in SELECT-OPTIONS or RANGES table

CHECK_SELOPT_ILLEGAL_SIGN

: Wrong " SIGN " in SELECT-OPTIONS or RANGES table

Department of Computer Science and Engineering

Subject Name: **ENTERPRISE SOLUTIONS**

Subject Code: **CS T61**

UNIT III

Oracle Suite : Oracle Apps 11i - Application Framework - File System - Workflow Analysis - SQL / PLSQL fundamentals - Creating Forms - Oracle Reports. Oracle Electronic Data Interchange – functions of EDI – Data File Structure - Oracle Data, Oracle Database - Oracle Database - DW vs OLTP - DW Connectors.

2 Marks

1. What is Oracle Application Architecture?

The *Oracle Applications Architecture* is a framework for multi-tiered, distributed computing that supports Oracle Applications products. In this model, various *servers* are distributed among multiple levels, or *tiers*.

2. What are the three tiers in an Oracle E-Business Suite installation?

- the *database tier*, which supports and manages the Oracle database
- the *application tier*, which supports and manages the various Applications components, and is sometimes known as the middle tier
- the *desktop tier*, which provides the user interface via an add-on component to a standard web browser

3. What is Web server?

The Oracle HTTP server (powered by Apache) acts as the Web server. It processes the requests received

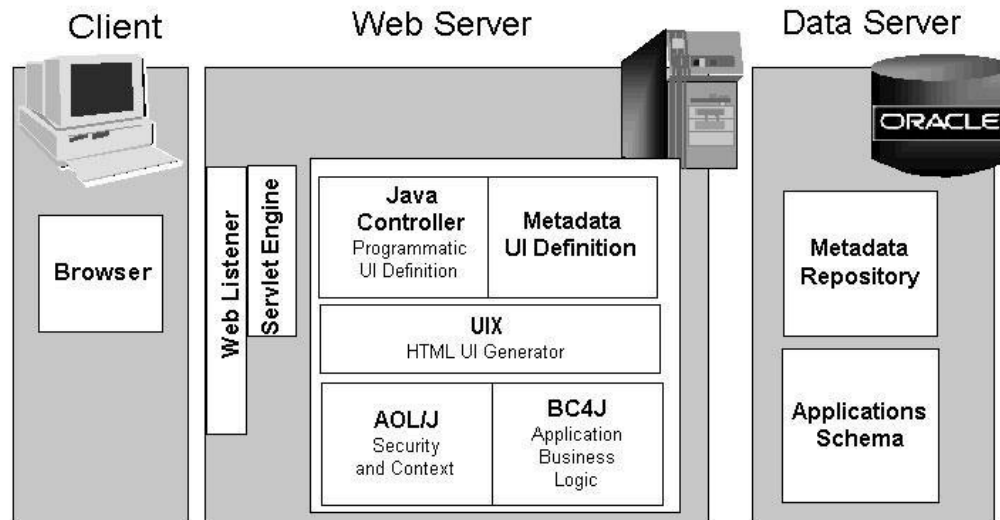
over the network from the desktop clients, and includes additional components such as:

- Web Listener
- Java Servlet Engine
- JavaServer Pages (JSP)

4. What is Oracle Application Framework?

The *Oracle Applications Framework* is the development platform for HTML-based applications. It consists of a Java-based application tier framework and associated services, designed to facilitate the rapid deployment of HTML-based applications.

5. Draw Oracle Applications Framework Architecture?



6. What is Discoverer Server?

Discoverer is an ad hoc query, reporting, analysis, and publishing tool that allows business users at all levels of an organization to gain immediate access to information from data marts, data warehouses, and online transaction processing (OLTP) systems.

7. What are operations from Admin server?

- Upgrading Oracle Applications
- Applying database patches to Oracle Applications
- Maintaining Oracle Applications data

8. What is Daily Business Intelligence(DBI)?

Daily Business Intelligence (DBI) is a reporting framework that is integrated with Oracle E-Business Suite. It replaces the Business Intelligence System (BIS), and includes a new set of materialized views that pre-summarize transaction data. Using Daily Business Intelligence *overview pages*, managers can view summarized information across multiple organizations, drilling down to specific transaction details on a daily basis.

9. What is oracle overflow?

Oracle Workflow provides an infrastructure for the enterprise-wide communication of data related to defined business events, providing the capabilities needed to:

- Manage enterprise business processes that may span trading partners
- Support standard and personalized business rules

- Streamline and automate transaction flows
- Manage exceptions without manual intervention

10.What is oracle workflow builder?

Oracle Workflow Builder provides a graphical drag and drop process designer. You can create and evolve business processes to incorporate existing business practices between your organization and customers or suppliers, without modifying existing business processes and without changing applications code.

11.What is the Oracle Workflow Business Event System?

It provides a workflow-enabled solution for your enterprise application integration requirements.

The Business Event System supports the following types of integration:

- Message-based point-to-point system integration
- System integration messaging hubs
- Distributed applications messaging

12.What is the Oracle Workflow Event Manager?

It enables registration of significant business events for selected applications, including functions that generate the XML event messages.

13.What are the features in Oracle E-Business Suite Release 11i?

Release 11i utilizes various Oracle database features to optimize performance, scalability, and business intelligence capacity.

14.What is the use cost-based optimization (CBO)?

The Oracle optimizer evaluates many factors to calculate the most efficient way to execute a SQL statement. The optimizer compares the costs of the execution plans and chooses the one with the smallest cost, i.e. optimum execution characteristics.

15.What are Materialized Views?

Materialized views are schema objects that can be used to summarize, precompute, replicate, and distribute data.

16.What is Real Application Clusters?

Real Application Clusters (RAC) harness the processing power of multiple interconnected computers. RAC software and a collection of computers (known as a cluster) harness the processing power of each component to create a robust and

powerful computing environment. A large task divided into subtasks and distributed among multiple nodes is completed more quickly and efficiently than if the entire task was processed on one node. Cluster processing also facilitates deployment of additional hardware resources for larger workloads and rapidly growing user populations.

17. Define Structured Query Language (SQL)?

An internationally standardized language that is used to access data in a relational database.

SQL statements enable you to perform the following tasks:

- Query data
- Insert, update, and delete rows in a table
- Create, replace, alter, and drop objects
- Control access to the database and its objects
- Guarantee database consistency and integrity

18. What is Data Definition Language (DDL) Statements?

Data definition language (DDL) statements define, structurally change, and drop schema objects. For example, DDL statements enable you to:

- Create, alter, and drop schema objects and other database structures, including the database itself and database users. Most DDL statements start with the keywords CREATE, ALTER, or DROP.
- Delete all the data in schema objects without removing the structure of these objects (TRUNCATE)
- Grant and revoke privileges and roles (GRANT, REVOKE).
- Turn auditing options on and off (AUDIT, NOAUDIT).
- Add a comment to the data dictionary (COMMENT).

19. What is Data Manipulation Language (DML) Statements?

DML statements are the most frequently used SQL statements and enable you to:

- Retrieve or fetch data from one or more tables or views (SELECT).
- Add new rows of data into a table or view (INSERT) by specifying a list of column values or using a subquery to select and manipulate existing data.
- Change column values in existing rows of a table or view (UPDATE).
- Update or insert rows conditionally into a table or view (MERGE).
- Remove rows from tables or views (DELETE)

20. What is PL/SQL?

PL/SQL provides a server-side, stored procedural language that is easy-to-use, seamless with SQL, robust, portable, and secure. You can access and manipulate database data using procedural schema objects called PL/SQL program units

21. List advantages of PL/SQL subprograms?

- Improved performance
- Memory allocation
- Improved productivity
- Security with definer's rights procedures

22.. What is PL/SQL Packages?

A PL/SQL package is a group of related subprograms, along with the cursors and use, stored together in the database for continued use as a unit.

23. Advantages of PL/SQL Packages?

- Encapsulation
- Data security
- Better Performance

24. What is Cursor?

A cursor is a pointer to this context area. PL/SQL controls the context area through a cursor. A cursor holds the rows (one or more) returned by a SQL statement. The set of rows the cursor holds is referred to as the **active set**.

25. What is Electronic Data Interchange (EDI)?

EDI is an electronic exchange of information between trading partners. Data files are exchanged in a standard format to minimize manual effort, speed data processing, and ensure accuracy.

26. What are the functions performed by EDI?

- define trading partner groups and trading partner locations
- enable transactions for trading partners
- provide location code conversion between trading partner location codes and codes used in Oracle Applications
- provide general code conversion between trading partner codes or standard codes
- define interface data files so that application data can interface with EDI translators

27. Write about Data File Structure?

- Oracle Database creates a data file for a table space by allocating the specified amount of disk space plus the overhead for the data file header.
- The operating system under which Oracle Database runs is responsible for clearing old information and authorizations from a file before allocating it to the database
- The data file header contains metadata about the data file such as its size and checkpoint SCN. Each header contains an absolute file number and a relative file number.
- The absolute file number uniquely identifies the data file within the database. The relative file number uniquely identifies a data file within a table space.

28. List some unstructured data?

- Spatial data
- Multimedia data
- Text data

29. What is LOB?

The large object (LOB) data types enable you to store and manipulate large blocks of unstructured data in binary or character format. LOBs provide efficient, random, piece-wise access to the data.

30. Write about external LOBs?

A BFILE(binary file LOB) is an external LOB because the database stores a pointer to a file in the operating system. The external data is read-only.

Overview of XML in Oracle Database

Oracle XML DB is a set of Oracle Database technologies related to high-performance XML storage and retrieval. XML DB provides native XML support by encompassing both SQL and XML data models in an interoperable manner

31. What is Oracle Spatial ?

Oracle Spatial makes spatial data management easier to users of location-enabled applications and geographic information system (GIS) applications

32. What is Oracle multimedia ?

Oracle Multimedia enables Oracle Database to store, manage, and retrieve images, medical images that follow the Digital Imaging and Communications in Medicine (DICOM) standard, audio, and video data in an integrated fashion with other enterprise information

33. Write advantages of Oracle Text?

- Oracle Text allows text searches to be combined with regular database searches in a single SQL statement. The Text index is in the database, and Text queries are run in the Oracle Database process. The optimizer can choose the best execution plan for any query, giving the best performance for ad hoc queries involving Text and structured criteria.

34. What is DBMS?

A database management system (DBMS) is software that controls the storage, organization, and retrieval of data. Typically, a DBMS has the following elements:

35. What is RDBMS?

- An RDBMS moves data into a database, stores the data, and retrieves it so that it can be manipulated by applications.
- Oracle Database is an RDBMS. An RDBMS that implements object-oriented features such as user-defined types, inheritance, and polymorphism is called an object-relational database management system (ORDBMS).

36. What is transaction management?

Oracle Database is designed as a multiuser database. The database must ensure that multiple users can work concurrently without corrupting one another's data.

37. What is meant by Data warehousing?

A data warehouse is a relational database designed for query and analysis rather than for transaction processing. For example, a data warehouse could track historical stock prices or income tax records. A warehouse usually contains data derived from historical transaction data, but it can include data from other sources.

38. List Characteristics of Data warehouse?

- Subject-Oriented
- Nonvolatile
- Time-Variant

39. Write Data warehouse Vs OLTP?

| Characteristics | Data Warehouse | OLTP |
|--------------------|--|--|
| Workload | Designed to accommodate ad hoc queries. You may not know the workload of your data warehouse in advance, so it should be optimized to perform well for a wide variety of possible queries. | Supports only predefined operations. Your applications might be specifically tuned or designed to support only these operations. |
| Data modifications | Updated on a regular basis by the ETL process using bulk data modification techniques. End users of a data warehouse do not directly update the database. | Subject to individual DML statements routinely issued by end users. The OLTP database is always up to date and reflects the current state of each business transaction. |
| Schema design | Uses denormalized or partially denormalized schemas (such as a star schema) to optimize query performance. | Uses fully normalized schemas to optimize DML performance and to guarantee data consistency. |
| Typical operations | A typical query scans thousands or millions of rows. For example, a user may request the total sales for all customers last month. | A typical operation accesses only a handful of records. For example, a user may retrieve the current order for a single customer. |
| Historical data | Stores many months or years of data to support historical analysis. | Stores data from only a few weeks or months. Historical data retained as needed to meet the requirements of the current transaction. |

11 Marks

1. Explain PL/SQL?

PL/SQL is Oracle's procedural language extension to SQL. PL/SQL combines the ease and flexibility of SQL with the procedural functionality of a structured programming language, such as **IF ... THEN**, **WHILE**, and **LOOP**.

When designing a database application, consider the following advantages of using stored PL/SQL:

- PL/SQL code can be stored centrally in a database. Network traffic between applications and the database is reduced, so application and system performance increases. Even when PL/SQL is not stored in the database, applications can send blocks of PL/SQL to the database rather than individual SQL statements, thereby reducing network traffic.
- Data access can be controlled by stored PL/SQL code. In this case, PL/SQL users can access data only as intended by application developers, unless another access route is granted.
- PL/SQL blocks can be sent by an application to a database, running complex operations without excessive network traffic.
- Oracle supports PL/SQL Server Pages, so your application logic can be invoked directly from your Web pages.

The following sections describe the PL/SQL program units that can be defined and stored centrally in a database.

PL/SQL Program Units

Program units are stored procedures, functions, packages, triggers, and autonomous transactions.

Procedures and **functions** are sets of SQL and PL/SQL statements grouped together as a unit to solve a specific problem or to perform a set of related tasks. They are created and stored in compiled form in the database and can be run by a user or a database application.

Procedures and functions are identical, except that functions always return a single value to the user. Procedures do not return values.

Packages encapsulate and store related procedures, functions, variables, and other constructs together as a unit in the database. They offer increased functionality (for example, global package variables can be declared and used by any procedure in the package). They also improve performance (for example, all objects of the package are parsed, compiled, and loaded into memory once).

2) Write Data Warehousing Vs OLTP?

| Characteristics | Data Warehouse | OLTP |
|---------------------|---|---|
| Source of data | Consolidation data; OLAP data comes from the various OLTP Databases | Operational data; OLTPs are the original source of the data. |
| Purpose of data | To help with planning, problem solving, and decision support | To control and run fundamental business tasks |
| What the data | Multi-dimensional views of various kinds of business activities | Reveals a snapshot of ongoing business processes |
| Inserts and Updates | Periodic long-running batch jobs refresh the data | Short and fast inserts and updates initiated by end users |
| Queries | Often complex queries involving aggregations | Relatively standardized and simple queries Returning relatively few records |
| Processing Speed | Depends on the amount of data involved; batch data refreshes and | Typically very fast |

| | | |
|---------------------|--|--|
| | complex queries may take many hours; query speed can be improved by creating indexes | |
| Space Requirements | Larger due to the existence of aggregation structures and history data; requires more indexes than OLTP | Can be relatively small if historical data is archived |
| Database Design | Typically de-normalized with fewer tables; use of star and/or snowflake schemas | Highly normalized with many tables |
| Backup and Recovery | Instead of regular backups, some environments may consider simply reloading the OLTP data as a recovery method | Backup religiously; operational data is critical to run the business, data loss is likely to entail significant monetary loss and legal liability |
| Workload | Designed to accommodate ad hoc queries. You may not know the workload of your data warehouse in advance, so it should be optimized to perform well for a wide variety of possible queries. | Supports only predefined operations. Your applications might be specifically tuned or designed to support only these operations. |
| Data modifications | Updated on a regular basis by the ETL process using bulk data modification techniques. End users of a data warehouse do not directly update the database. | Subject to individual DML statements routinely issued by end users. The OLTP database is always up to date and reflects the current state of each business transaction. |
| Schema design | Uses denormalized or partially denormalized schemas (such as a star schema) to optimize query performance. | Uses fully normalized schemas to optimize DML performance and to guarantee data consistency. |
| Typical operations | A typical query scans thousands or millions of rows. For example, a user may request the total sales for all customers last month. | A typical operation accesses only a handful of records. For example, a user may retrieve the current order for a single customer. |
| Historical data | Stores many months or years of data to support historical analysis. | Stores data from only a few weeks or months. Historical data retained as needed to meet the requirements of the current transaction. |

3) Write about Oracle Data?

The traditional relational model deals with simple structured data that fits into simple tables. Oracle Database also provides support for unstructured data, which cannot be decomposed into standard components. Unstructured data includes text, graphic images, video clips, and sound waveforms.

Oracle Database includes data types to handle unstructured content. These data types appear as native types in the database and can be queried using SQL

Unstructured Data:

Oracle Spatial Data:

Oracle Spatial makes spatial data management easier to users of location-enabled applications and geographic information system (GIS) applications.

Oracle Multimedia Data:

Oracle Multimedia enables Oracle Database to store, manage, and retrieve images, medical images that follow the Digital Imaging and Communications in Medicine (DICOM) standard, audio, and video data in an integrated fashion with other enterprise information. Oracle Multimedia provides object types and methods for:

Oracle Text Data:

Oracle Text (Text) is a fast and accurate full-text retrieval technology integrated with Oracle Database. Oracle Text indexes any document or textual content stored in file systems, databases, or on the Web. These documents can be searched based on their textual content, metadata, or attributes.

LOBs

The large object (LOB) data types enable you to store and manipulate large blocks of unstructured data in binary or character format. LOBs provide efficient, random, piece-wise access to the data.

Internal LOBs An internal LOB stores data in the database itself rather than in external files. Internal LOBs include the following:

- CLOB(character LOB), which stores large amounts of text, such as text or XML files, in the database character set
- NCLOB(national character set LOB), which stores Unicode data
- BLOB(binary LOB), which stores large amounts of binary information as a bit stream and is not subject to character set translation

External LOBs A BFILE(binary file LOB) is an external LOB because the database stores a pointer to a file in the operating system. The external data is read-only.

Overview of XML in Oracle Database

Oracle XML Db is a set of Oracle Database technologies related to high-performance XML storage and retrieval. XML DB provides native XML support by encompassing both SQL and XML data models in an interoperable manner

4)_Write about Electronic Data Interchange (EDI)?

Oracle Applications provides users with the ability to conduct business electronically between trading partners based on the Electronic Commerce standards and methodology. One form of Electronic Commerce is Electronic Data Interchange (EDI).

EDI is an electronic exchange of information between trading partners. Data files are exchanged in a standard format to minimize manual effort, speed data processing, and ensure accuracy.

The EDI Gateway performs the following functions:

- define trading partner groups and trading partner locations
- enable transactions for trading partners
- provide location code conversion between trading partner location codes and codes used in Oracle Applications
- provide general code conversion between trading partner codes or standard codes
- define interface data files so that application data can interface with EDI translators
- extract application data, format, and write to data files (outbound transactions)
- import data or converted codes into application open interface tables so that application program interfaces (API) can validate and update Oracle application tables (inbound transactions)

How Oracle EDI Gateway Works with Other Oracle Applications

Oracle Applications are designed with an open architecture for easy integration with EDI translators and electronic transmission products to provide a seamless solution. Oracle Applications utilize the Oracle EDI Gateway to integrate with EDI translator software. EDI translation software packages integrate with an electronic transmission service to provide a closed-loop between Oracle Applications and the trading partner's application.

The Oracle Applications for Manufacturing, Distribution, and Financials are EDI-enabled using the Oracle EDI Gateway product. The Oracle EDI Gateway product augments the existing standard

paper document capabilities of Oracle Applications, or adds functionality where no corresponding paper documents exist.

A common EDI implementation is via ASCII data files in a batch environment. Data from the sending application is extracted into an application data file. The application data file is received by the translation software which translates it into the an EDI standard both trading partners agree upon. Then the EDI data file is placed on a network for transmission to the receiving application. The receiving application's EDI translator receives the EDI data file from the network and begins the file processing in reverse sequence. The translator translates the EDI data file and creates an application data file meaningful to the receiving application. The receiving application receives the application data file for processing and imports the data into the application.

The following figure illustrates the outbound EDI Gateway transaction flow:

5)Write about Oracle workflow analysis?

Oracle Workflow provides an infrastructure for the enterprise-wide communication of data related to your organization and customers or suppliers, without modifying existing business processes and without changing applications code. The Oracle Workflow *Business Event System* provides a workflow-enabled solution for your enterprise application integration requirements. The Business Event System is an application service delivered with Oracle Workflow that uses Oracle Advanced Queuing technology to communicate business events between systems. The Business Event System supports the following types of integration:

- Message-based point-to-point system integration
- System integration messaging hubs
- Distributed applications messaging

The Business Event System uses Oracle Advanced Queuing to propagate messages between communication points on systems, called *agents*, using a specified protocol. Events received from external systems are processed by an agent listener that runs on that agent's queue.

The *Oracle Workflow Event Manager* enables registration of significant business events for selected applications, including functions that generate the XML event messages. Users of those applications can register *subscriptions* on events that are significant to their systems, to take actions such as triggering custom code.

Operation

When a business event occurs, the Workflow Event Manager executes any subscriptions registered on the event. For local events, the subscribing code can be executed synchronously, in the same database transaction as the code that raised the event, or asynchronously, deferring costly subscription processing to a later time, and thus allowing control to be returned more quickly to the calling application. Events can also be received asynchronously from external systems. Before producing the XML event message, the Event Manager minimizes processing by checking whether event information is required by subscriptions on the event.

You can review and respond to your business process notifications from one central window, known as the worklist, using a standard Web browser. This offers the flexibility to prioritize tasks and to define sort criteria, giving you the flexibility to organize your work the way you wish. For example, you can group notifications by type or subject, to avoid having to jump from one context to another. Alternatively, you can focus on time critical tasks first, sorting by priority or due date. Oracle Workflow is fully integrated with the Oracle E-Business Suite, providing the ability to drill down to any Oracle E-Business Suite or associated URL to view or complete a transaction.

Additional Capabilities

The flexibility of the powerful *Workflow Engine* event activities enable you to model business events within workflow processes. Event activities can be used to model content-based routing, transformations, error handling, and so on. A workflow process can be started or processed by an inbound message, and can send an outbound message or raise an event to the Event Manager. XML function activities give you access to event content data within workflow processes. Workflow processes based on business events give the greatest flexibility when implementing an integration solution. However, the Business Event System can also run independently of the Workflow Engine, to enable point-to-point messaging to be utilized.

You can perform complex transformations between different formats required for your business documents. Oracle Workflow allows you to apply a style sheet to an XML event message. In addition, when queues are defined within the Business Event System, you specify the logic used to enqueue and dequeue messages. This logic, called a *queue handler*, can include transformations.

Oracle Workflow also allows you to take advantage of XML support in the Oracle database. Oracle9i and later releases deliver native support for XML via a new XML data type, which makes the manipulation of XML data and documents in applications seamless and straightforward. From Oracle9i, the database server offers the capability to generate, massage, and transform XML data

and documents inside the runtime engine itself, giving excellent scalability and performance. Defined business events, providing the capabilities needed to:

- Manage enterprise business processes that may span trading partners
- Support standard and personalized business rules
- Streamline and automate transaction flows
- Manage exceptions without manual intervention

Oracle Workflow lets you model and maintain your business processes using a graphical workflow builder. You can model and automate sophisticated business processes, defining processes that can loop, branch into parallel flows and rendezvous, decompose into sub-flows, branch on task results, time out, and more.

Acting as a *system integration hub*, Oracle Workflow can apply business rules to control objects and route them between applications and systems. It extends the reach of business process automation throughout an enterprise and beyond, to include any email user, web user, or system, enabling people to receive, analyze, and respond to *notifications* needing their attention. Users can respond to a notification via any standard email system or standard Web browser.

Components

Oracle Workflow Builder provides a graphical drag and drop process designer.

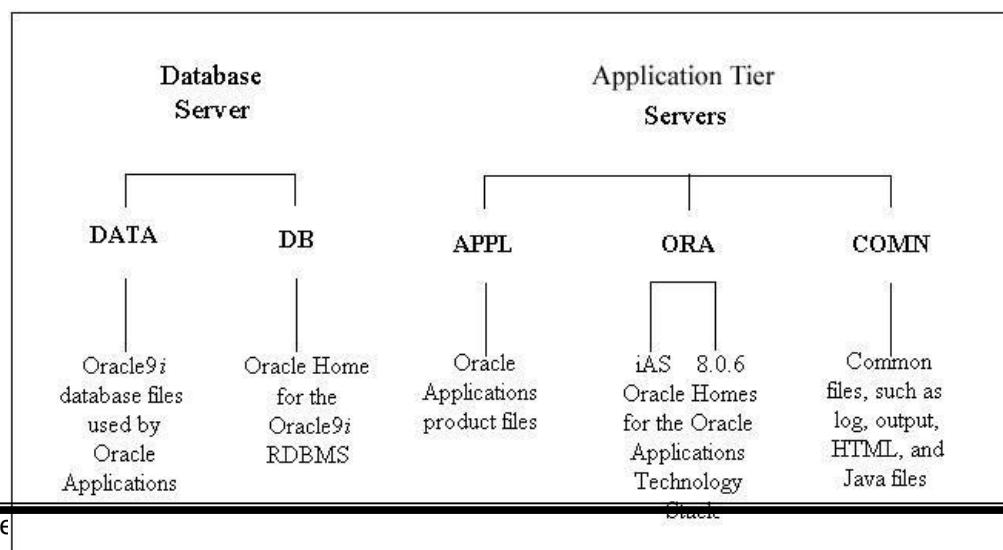
6)_Write Oracle 11i File System

Introduction

An Oracle Applications Release 11i system utilizes components from many Oracle products. These product files are stored below a number of key top-level directories on the database and application server machines.

Operating system *environment settings* indicate the location of the various files in the file systems of the database and application server machines.

Applications Directory Structure



Oracle Applications Environment

- Oracle Applications makes extensive use of environment settings to locate executable programs and other files essential to Applications operation
- The environment settings and their associated values are stored in *environment files*, which have a *.env* suffix (*.cmd* on Windows)

The DATA Directory

The *<dbname>*DATA file system contains the data (.dbf) files of the Oracle database.

DB and ORA Directories

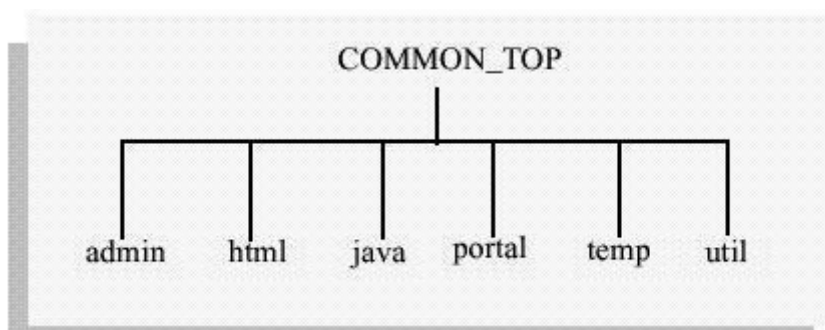
Release 11*i* utilizes three ORACLE_HOMEs:

- The 9.2.0 ORACLE_HOME (Applications database home) is located in the *<dbname>*DB directory. It contains **the files needed for running and maintaining the Oracle Applications database.**
- The 8.0.6 directory contains the ORACLE_HOME for the Developer 6*i* products (Forms, Reports, and Graphics). The product libraries in the 8.0.6 ORACLE_HOME are **used to relink Oracle Applications executables.**
- The iAS directory, also under the *<dbname>*ORA, contains the ORACLE_HOME **for Oracle9*i* Application Server.**

The COMN Directory

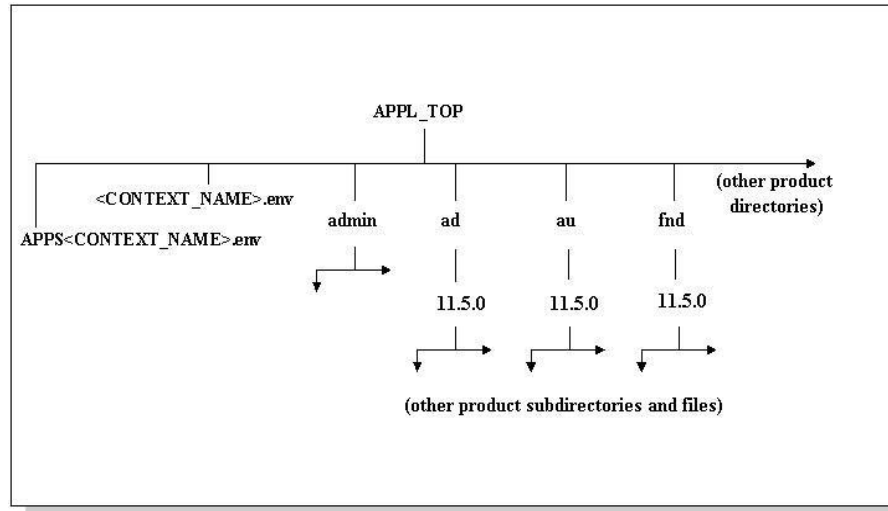
The *<dbname>*COMN or COMMON_TOP directory contains files used by many different Oracle Applications products, and which may also be used with third-party products.

COMMON_TOP Directory Structure



The APPL Directory

Oracle Applications files are stored in the `<dbname>APPL` directory, which is known as the *APPL_TOP* directory ***APPL_TOP Directory Structure***

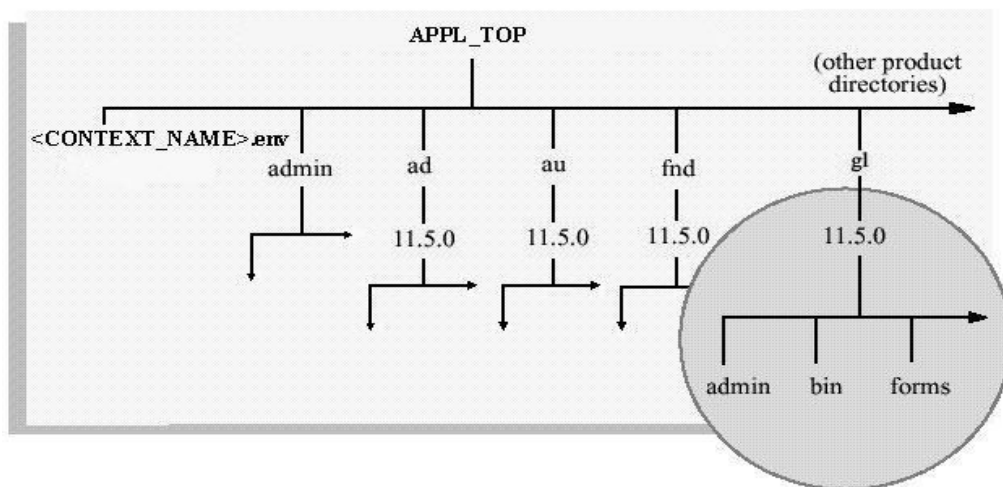


Product Directories

<PROD>_TOP Directory

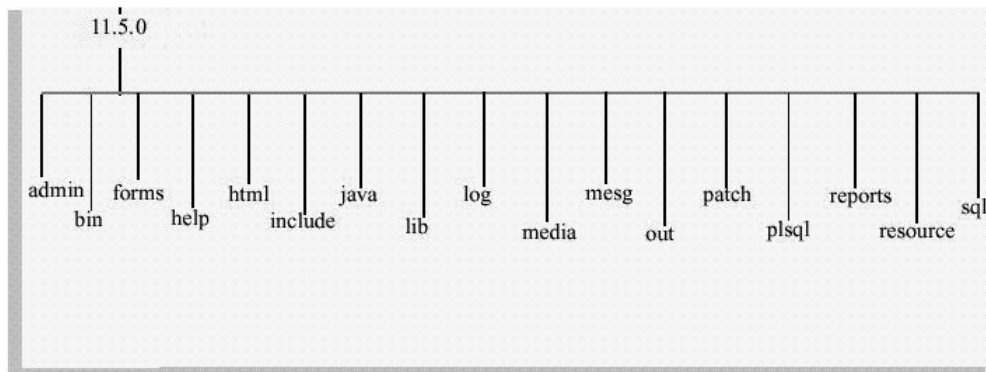
The `<APPL_TOP>/<prod>/<version>` path is known as the *product top directory* (`<PROD>_TOP`), and its value is stored in the `<PROD>_TOP` environment variable.

APPL_TOP Directory Structure



Within each `<PROD>_TOP` directory, the product's files are grouped into subdirectories according to file type and function. The next figure expands the inset to show the full directory structure for *gl*.

Detail of gl Directory Structure



Language Files

When you install Oracle Applications in a language other than American English, each product tree includes directories that use the NLS language code.

Core Technology Directories

The *admin*, *ad*, *au*, and *fnd* directories are the core technology directories.

Distributing the APPL_TOP Across Disks

The Oracle Applications file system on the application tier requires a significant amount of disk space

Other Environment Files

Several other key environment files are used in an Oracle Applications system.

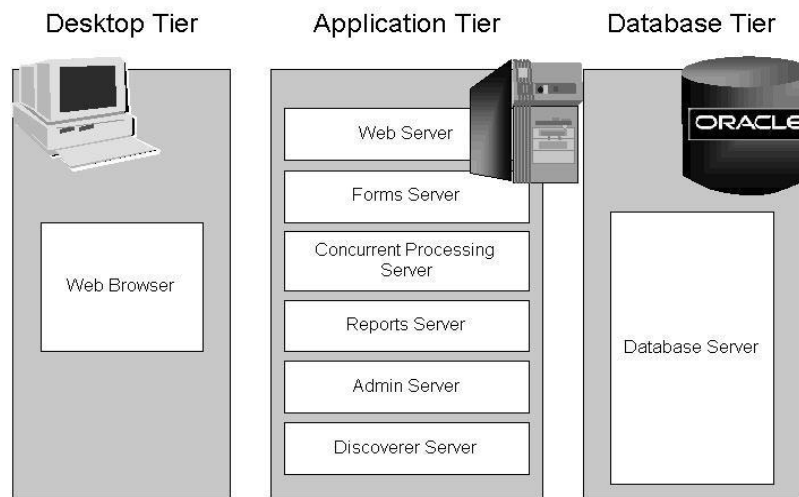
The adovars.env file

The *adovars.env* file, located in `$APPL_TOP/admin`, specifies the location of various files such as Java files, HTML files, and JRE (Java Runtime Environment) files. It is called from the main applications environment file, `<CONTEXT_NAME>.env`.

7. Write about Oracle Framework?

- The *Oracle Applications Architecture* is a framework for multi-tiered, distributed computing that supports Oracle Applications products. In this model, various *servers* are distributed among multiple levels, or *tiers*.
- A server is a process or group of processes that runs on a single machine and provides a particular functionality, often referred to as a *service*.
- For example, the *HTTP server* is a process that listens for and processes HTTP requests, and the *Forms server* is a process that listens for and processes requests for activities related to Oracle Forms.

- A tier is a logical grouping of services, potentially spread across more than one physical machine.
- The three-tier architecture that comprises an Oracle E-Business Suite installation is made up of
 - the *database tier*, which supports and manages the Oracle database;
 - the *application tier*, which supports and manages the various Applications components, and is sometimes known as the middle tier; and
 - the *desktop tier*, which provides the user interface via an add-on component to a standard web browser.



The Desktop Tier

The client interface is provided through HTML for the newer HTML-based applications, and via a Java applet in a Web browser for the traditional Forms-based interface.

- In Oracle Applications Release 11i, each user logs in to Oracle Applications through the E-Business Suite Home Page on a desktop client web browser.
- The E-Business Suite Home Page provides a single point of access to HTML-based applications, Forms-based applications, and Business Intelligence application

The Application Tier

- The *application tier* has a dual role: hosting the various servers that process the business logic, and managing communication between the desktop tier and the database tier. This tier is sometimes referred to as the *middle tier*.

HTML-Based Applications and the Oracle Applications Framework

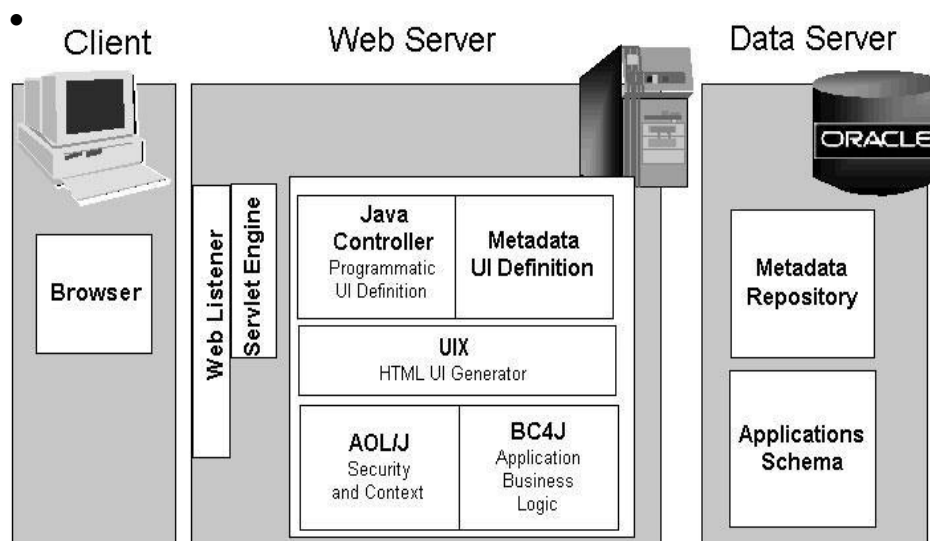
The Oracle HTML-based (formerly known as Self-Service) Applications:

- Do not use Oracle Forms for the interface

- Are designed in pure HTML and JavaScript
- Dynamically generate HTML pages by executing Java code
- Use a metadata dictionary for flexible layout
- Operate by direct connection to the Web server
- The *Oracle Applications Framework* is the development platform for HTML-based applications.
- It consists of a Java-based application tier framework and associated services, designed to facilitate the rapid deployment of HTML-based applications.

Notable Oracle Applications Framework components include:

- **Business Components for Java (BC4J)**, included in Oracle JDeveloper, is used to create Java business components for representing business logic. *AOL/J* supplies the Oracle Applications Framework with underlying security and applications Java services.



Oracle Applications Framework Processing Details

The following is a more detailed explanation of how the JSP obtains the content from the Applications tables and uses information from the metadata dictionary to construct the HTML page.

- AOL/J validates user access to the page.
- The page definition (metadata UI definition) is loaded from the metadata repository on the database tier into the application tier.
- The BC4J objects that contain the application logic and access the database are instantiated.
- The Java Controller programmatically manipulates the page definition as necessary, based on dynamic UI rules.

- UIX (HTML UI Generator) interprets the page definition, creates the corresponding HTML in accordance with UI standards, and sends the page to the browser.

8) Explain Oracle Database?

About Relational Databases

Every organization has information that it must store and manage to meet its requirements. For example, a corporation must collect and maintain human resources records for its employees. This information must be available to those who need it. An information system is a formal system for storing and processing information.

An information system could be a set of cardboard boxes containing manila folders along with rules for how to store and retrieve the folders. However, most companies today use a database to automate their information systems. A database is an organized collection of information treated as a unit. The purpose of a database is to collect, store, and retrieve related information for use by database applications.

Database Management System (DBMS)

A database management system (DBMS) is software that controls the storage, organization, and retrieval of data. Typically, a DBMS has the following elements:

The first generation of database management systems included the following types:

■ **Hierarchical**

A hierarchical database organizes data in a tree structure. Each parent record has one or more child records, similar to the structure of a file system.

■ **Network**

A network database is similar to a hierarchical database, except records have a many-to-many rather than a one-to-many relationship.

Relational Model

A relational database is a database that conforms to the relational model.

The relational model has the following major aspects:

■ **Structures**

Well-defined objects store or access the data of a database.

■ **Operations**

Clearly defined actions enable applications to manipulate the data and structures of a database.

■ Integrity rules

Integrity rules govern operations on the data and structures of a database. A relational database stores data in a set of simple relations. A relation is a set of tuples. A tuple is an unordered set of attribute values.

A table is a two-dimensional representation of a relation in the form of rows (tuples) and columns (attributes). Each row in a table has the same set of columns. A relational database is a database that stores data in relations (tables). For example, a relational database could store information about company employees in an employee table, a department table, and a salary table.

Relational Database Management System (RDBMS)

The relational model is the basis for a relational database management system (RDBMS). Essentially, an RDBMS moves data into a database, stores the data, and retrieves it so that it can be manipulated by applications

Schema Objects

One characteristic of an RDBMS is the independence of physical data storage from logical data structures. In Oracle Database, a database schema is a collection of logical data structures, or schema objects. A database schema is owned by a database user and has the same name as the user name.

Data Access

A general requirement for a DBMS is to adhere to accepted industry standards for a data access language.

Structured Query Language (SQL)

SQL is a set-based declarative language that provides an interface to an RDBMS such as Oracle Database.

A SQL statement is a string of SQL text such as the following:

```
SELECT first_name, last_name FROM employees;
```

SQL statements enable you to perform the following tasks:

- Query data
- Insert, update, and delete rows in a table
- Create, replace, alter, and drop objects
- Control access to the database and its objects
- Guarantee database consistency and integrity

SQL unifies the preceding tasks in one consistent language. Oracle SQL is an implementation of the ANSI standard. Oracle SQL supports numerous features that extend beyond standard SQL.

PL/SQL and Java

PL/SQL is a procedural extension to Oracle SQL. PL/SQL is integrated with Oracle Database, enabling you to use all of the Oracle Database SQL statements, functions, and data types. You can use PL/SQL to control the flow of a SQL program, use variables, and write error-handling procedures.

Transaction Management

Oracle Database is designed as a multiuser database. The database must ensure that multiple users can work concurrently without corrupting one another's data.

Transactions

An RDBMS must be able to group SQL statements so that they are either all committed, which means they are applied to the database, or all rolled back, which means they are undone. **Data**

Concurrency

A requirement of a multiuser RDBMS is the control of concurrency, which is the simultaneous access of the same data by multiple users. Oracle Database uses locks to control concurrent access to data.

Data Consistency

Oracle Database always enforces statement-level read consistency, which guarantees that the data returned by a single query is committed and consistent with respect to a single point in time.

9) Explain Data File Structure?

An Oracle database is a set of files that store Oracle data in persistent disk storage. This section discusses the database files generated when you issue a CREATE DATABASE statement:

- Data files and temp files

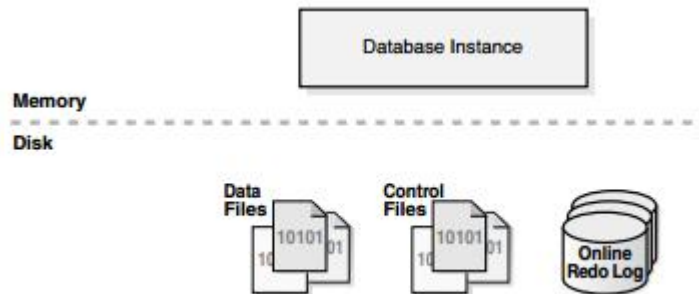
A data file is a physical file on disk that was created by Oracle Database and contains data structures such as tables and indexes. A temp file is a data file that belongs to a temporary table space. The data is written to these files in an Oracle proprietary format that cannot be read by other programs.

- Control files

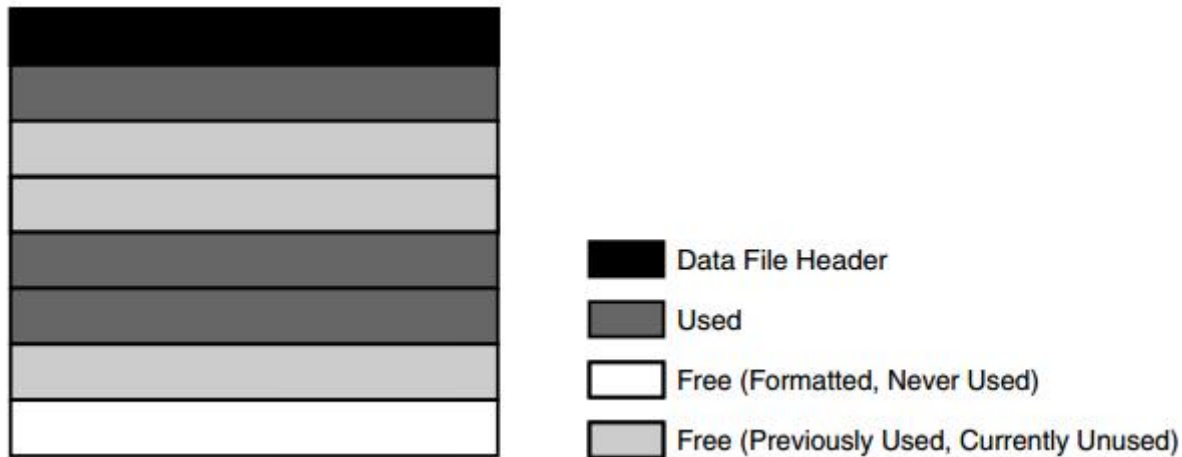
A control file is a root file that tracks the physical components of the database.

- Online redo log files

The online redo log is a set of files containing records of changes made to data



- Oracle Database creates a data file for a table space by allocating the specified amount of disk space plus the overhead for the data file header.
- The operating system under which Oracle Database runs is responsible for clearing old information and authorizations from a file before allocating it to the database
- The data file header contains metadata about the data file such as its size and checkpoint SCN. Each header contains an absolute file number and a relative file number
- The absolute file number uniquely identifies the data file within the database. The relative file number uniquely identifies a data file within a table space.
- When Oracle Database first creates a data file, the allocated disk space is formatted but contains no user data. However, the database reserves the space to hold the data for future segments of the associated table space.
- As the data grows in a table space, Oracle Database uses the free space in the data files to allocate extents for the segment.
- Figure illustrates the different types of space in a data file. Extents are either used, which means they contain segment data, or free, which means they are available for reuse. Over time, updates and deletions of objects within a table space can create pockets of empty space that individually are not large enough to be reused for new data. This type of empty space is referred to as fragmented free space.



Space in a Data File

10) Explain SQL?

SQL is the set-based, high-level declarative computer language with which all programs and users access data in an Oracle database

Oracle SQL statements are divided into the following categories:

- Data Definition Language (DDL) Statements
- Data Manipulation Language (DML) Statements
- Transaction Control Statements

Data Definition Language (DDL) Statements

Data definition language (DDL) statements define, structurally change, and drop schema objects.

For example, DDL statements enable you to:

- Create, alter, and drop schema objects and other database structures, including the database itself and database users. Most DDL statements start with the keywords CREATE, ALTER, or DROP.
- Delete all the data in schema objects without removing the structure of these objects (TRUNCATE)
- Grant and revoke privileges and roles (GRANT, REVOKE).
- Turn auditing options on and off (AUDIT, NOAUDIT).
- Add a comment to the data dictionary (COMMENT).

Data Manipulation Language (DML) Statements

Data manipulation language (DML) statements query or manipulate data in existing schema objects. Whereas DDL statements enable you to change the structure of the database, DML statements

enable you to query or change the contents. For example, ALTER TABLE changes the structure of a table, whereas INSERT adds one or more rows to the table.

DML statements are the most frequently used SQL statements and enable you to:

- Retrieve or fetch data from one or more tables or views (SELECT).
- Add new rows of data into a table or view (INSERT) by specifying a list of column values or using a subquery to select and manipulate existing data.
- Change column values in existing rows of a table or view (UPDATE).
- Update or insert rows conditionally into a table or view (MERGE).
- Remove rows from tables or views (DELETE).

Example:

```
SELECT * FROM employees;
```

```
INSERT INTO employees (employee_id, last_name, email, job_id, hire_date, salary)
VALUES (1234, 'Mascis', 'JMASCIS', 'IT_PROG', '14-FEB-2011', 9000);
```

```
UPDATE employees SET salary=9100 WHERE employee_id=1234;
```

```
DELETE FROM employees WHERE employee_id=1234;
```

Transaction Control Statements

Transaction control statements manage the changes made by DML statements and group DML statements into transactions.

- Make changes to a transaction permanent (COMMIT).
- Undo the changes in a transaction, since the transaction started (ROLLBACK) or since a savepoint (ROLLBACK TO SAVEPOINT). A savepoint is a user-declared intermediate marker within the context of a transaction

11) Explain Oracle Database 11i Feature?

Introduction

Many features in Oracle E-Business Suite Release 11i are built on the advanced capabilities of the underlying Oracle database technology. Release 11i utilizes various Oracle database features to **optimize performance, scalability, and business intelligence capacity.**

Performance Features

Database performance features include **optimization, resource usage, space management, and access rights.**

Cost-Based Optimization

Using CBO, the optimizer considers the available access paths, factoring in statistical information for the tables and indexes that the SQL statement will access.

For some operations, such as batch processing, Release 11i uses CBO to achieve the most efficient means of processing *all rows* that are accessed by the statement.

Database Resource Manager

The gives the system administrator extensive control over processing resources on the database node. The administrator can distribute server CPU based on business rules, ensuring that the highest priority activities always have sufficient CPU resources.

Partitioned Tables

Partitioning helps support very large tables and indexes by dividing them into smaller, more manageable pieces called *partitions*. Once the desired partitions have been defined, SQL statements can access them instead of the original tables or indexes.

Business Intelligence Features

To meet the increasing demand for up-to-date details of business activities, Oracle Applications utilizes Oracle database features that help to optimize the types of query typically required in such environments

Materialized Views

Materialized views are schema objects that can be used to summarize, precompute, replicate, and distribute data. They can markedly increase the speed of queries on very large databases when used to precompute and store aggregated data such as sums and averages.

Scalability

As well as providing more computing power, multi-node systems facilitate the addition of machines to meet increases in demand. They also provide resilience in the event of failures of individual components.

Real Application Clusters: Real Application Clusters (RAC) harness the processing power of multiple interconnected computers. RAC software and a collection of computers (known as a *cluster*) harness the processing power of each component to create a robust and powerful computing environment.

12) How to develop forms in Oracle?

Tutorial Connect Oracle Forms 11g to database 11g

1)First you need to copy the file tnsnames.ora from database folder to config folder from Oracle Middleware.

2) Open the Oracle Form Builder, go to File Menu and press Connect or just execute the command Ctrl+J.

3) On the last step, you need to enter the user name, password and the database name.

4) Change Module Name in Oracle Forms 11g

Right-click on Form Module name, then choose Property Palette. On Property Palette just click on the name and write your module name.

5) Open the Oracle Form Builder, go to File Menu and press Connect or just execute the command Ctrl+J.

6) Create New Data Block Wizard in Oracle Forms 11g

First you must click on Data Blocks and then press the button Create. An alert with two options will appear, and here you choose: Use the Data Block Wizard.

7) Welcome to the Data Block Wizard! - Click Next to begin creating your data block.

8) Select the type of Data Block: Table/View or Stored Procedure.

9) Create New Data Block Wizard in Oracle Forms 11g

Press Browse button and will appear a small window with more choices: Tables, Views, Current user, Synonyms, Other users.

10) After you choose the table name, Oracle Forms will show all Available Columns from the table. Here you can bring all columns from the table in your form or you can select just a few columns.

11) After you moved the items press Next.

12) Create New Data Block Wizard in Oracle Forms 11g

Enter a name for your Data Block.

13) You have finished describing your data block. Now you will have two options:

- Continue with: Call the Layout Wizard
- Stop: Just create the data block (is recommended when you use multiple data block on the same canvas)

13) Create Wizard Layout Oracle Forms 11g: Display in a frame on a canvas

14) Here you have to choose the canvas name (can be a new one or an existing canvas) and the type of canvas.

Types of canvas:

- Content
- Stacked
- Vertical Toolbar

- Horizontal Toolbar

- Tab

15) Create Wizard Layout Oracle Forms 11g

Here you select and move item that you want to be displayed in the frame.

For each item displayed you can choose the item type from the list.

16) Enter a prompt, width and the height for each item.

17) Create Wizard Layout Oracle Forms 11g

Select the Layout Style for your frame

18) On this step you can choose the Frame Title, you can insert how many records to be displayed in the field Records Displayed, also Distance Between Records and Display Scrollbar.

19) Finish the Wizard Layout and display the new frame in the Layout Editor

You Finish the Wizard Layout.

20) Display the new frame in the Layout Editor.

13) How to generate Report?

1) Connect Oracle Reports 11g to database 11g

Open the Oracle Reports Builder, go to File Menu and press Connect or execute the command Ctrl+J.

2) On the second step, you need to enter the user name, password and the database name.

Create Report Wizard in Oracle Reports 11g

First you must click on Reports and the press the button Create. An alert "Create a new report" with two option will appear, and here you choose: Use the Report Wizard.

3) Welcome to the Report Wizard! - Click Next to begin creating your report.

4) Choose The Type of Layout and Style Reports in Oracle 11g

Here you must choose the type of layout you would like to generate: Create both Web and Paper Layout, Create Web Layout only, Create paper Layout only.

5) On the next step you have to choose the report style and give a title name for your report. The report style option are: Tabular, Group Left, Group Above, Matrix, Matrix with Group.

6) Choose a Data Source and Select Sql Query Builder for Reports in Oracle 11g

On this step you have to choose a Data Source for your report. The Data Source can be from JDBC Query, SQL Query, Text Query, XML Query.

7) You can do write the query manually, you can use Query Builder or Import Builder.

8) Move Available Fields in Displayed Fields in R reports Oracle 11g

Select the fields that you would like to display in your report. You can move one item or all items to target from Available Fields in Displayed Fields.

9) After you moved the items press Next.

10) Select Available Fields in Reports Wizard to calculate totals

Select the fields that you would like to calculate totals in your report. Also you can do choose to calculate: sum, count, minimum, maximum, average.

11) Here you can modify the labels of the fields and also the width for each field.

12) Choose a Template for your Report Wizard in Oracle Reports 11g

Select the template for your report. You have multiple options: Predefined Template, Template File or No Template.

13) You have finished describing your report!

14) Run Module Report Editor Paper Design Oracle Reports 11g

Select the template for your report. You have multiple options: Predefined Template, Template File or No Template.

14. Explain any two Data warehouse Connectors?**1) Teradata OLAP Connector**

Teradata OLAP Connector is the one tool available that makes the direct connection between the Teradata system and Excel applications.

Overview

Teradata OLAP Connector provides self-service analytics for Excel, Tableau, and arcplan users by enabling them to access the corporate data stored in the Teradata Data Warehouse. This eliminates the inconsistencies and security concerns associated with the traditional approach of extracting and storing corporate data locally.

Challenges

Desktop-based BI solutions allow business users to extract data from the data warehouse for desktop analysis but this approach is challenging:

- Creates spread marts all over the organization
- Static view of the data of when it was extracted which leads to inconsistencies with the reporting

- Limited data leads to limited analysis
- Causes security challenges for leaving sensitive data on the users' desktops
- Data extractions scripts are difficult to maintain

ORACLE BIG DATA CONNECTORS

Oracle Big Data Connectors is a software suite that integrates processing in Hadoop with operations in a data warehouse. Designed to leverage the latest features of Apache Hadoop, Big Data Connectors connect Hadoop clusters with database infrastructure to harness massive volumes of structured and unstructured data for critical business insights. Big Data Connectors greatly simplify development and are optimized for efficient connectivity and high-performance between Oracle Big Data Appliance and Oracle Exadata. Oracle Big Data Connectors 3.0 delivers a rich set of new features, increased connectivity, enhanced performance, and security for Big Data applications.

Oracle Big Data Connectors

Large volumes of data are increasingly collected and processed in Hadoop, while enterprise IT systems are centered on relational data warehouses. Oracle Big Data Connectors bridges data processing in Hadoop with Oracle Database, providing the crucial ability to unify data across these systems. Combining pre-processing of large data volumes of raw and unstructured data in Hadoop with the advanced analytics, complex data management, and real-time query capabilities of Oracle Database, Oracle Big Data Connectors deliver features that support information discovery, deep analytics and fast integration of all data in the enterprise. The components of this software suite are:

Oracle SQL Connector for Hadoop Distributed File System

Oracle Loader for Hadoop

Oracle Data Integrator Application Adapter for Hadoop

Oracle R Advanced Analytics for Hadoop

Oracle XQuery for Hadoop

Oracle Big Data Connectors work with Oracle's engineered systems - Oracle Big Data Appliance and Oracle Exadata - as well as with supported Hadoop distributions and database versions on non-engineered systems.

KEY FEATURES

- Tight integration with Oracle Database
- Leverage Hadoop compute resources for data in HDFS

- Enable Oracle SQL to access and load Hadoop data
- Fast and very efficient load from Hadoop into Oracle Database
- Partition pruning of Hive tables during load and query
- Graphical user interfaces of Oracle Data Integrator drive data transformation workflows on

Department of Computer Science and EngineeringSubject Name: **ENTERPRISE SOLUTIONS**Subject Code: **CS T61****UNIT – IV**

PeopleSoft: Basic PeopleSoft Functionality – Opening Multiple Windows - Database structure – Understanding People Soft Data Mover – Records - Pages vs. Forms. PeopleSoft HRMS: Introduction to PeopleSoft HRMS database - PeopleSoft products - Functional PeopleSoft - financial management system - PeopleSoft Enterprise HRMS.

2 MARKS**1. What is Peoplesoft?**

PeopleSoft, Inc. was a company that provided Human Resource Management Systems (**HRMS**), Financial Management Solutions (**FMS**), Supply Chain Management (**SCM**), Customer Relationship Management(**CRM**), and Enterprise Performance Management (**EPM**) software, as well as software for manufacturing, and student administration to large corporations, governments, and organizations.

2. When Peoplesoft is founded?

Founded in **1987** by **Ken Morris** and **David Duffield**, PeopleSoft was originally headquartered in Walnut Creek, California before moving to Pleasanton, California. Duffield envisioned a client-server version of Integral Systems' popular mainframe HRMS package.

3. Which corporation control Peoplesoft?

It existed as an independent corporation until its acquisition by **Oracle Corporation** in 2005. The PeopleSoft name and product line are now marketed by Oracle.

4. What are the applications of Peoplesoft?

The applications are comprehensive business and industry solutions, enabling organizations to increase productivity, accelerate business performance, and provide a lower cost of ownership.

5. What are the Benefits of Peoplesoft?

- End-user initiate transactions and date in to the system once and make it accessible for multiple purposes (Finance, Human Resources, Reports)
- Tracking of transaction status is available online, almost eliminating the need to make multiple phone calls regarding the transaction's status.
- Various transactions processing are online
- Transaction approvals have been simplified

- Some transactions can be approved and submitted online, while meeting internal control, and presenting less need to hand carry papers from office to office.

6. Define Multiple Windows?

IE 8 can open multiple unique sessions (but only when using the new IE 8 menu option called "New Session" or when creating a new shortcut with a special -no merge instruction sent to IE executable) - do not use Windows Start > Internet Explorer if you want a new session - IE 7 can open as well new unique sessions but each time one should use the Windows Start > Internet Explorer to start a new browser instance (this is opposite to IE 8 new behavior - see above) - Using New Window (from any IE version or Firefox ...from menu or using CTRL+N ...or from PIA "New Window" links) will always open a new window tied to the same browser session - this will share the same cookies with previous windows.

7. Define Data Mover?

PeopleSoft Data Mover, is used to import / export data between databases. Apart from this, Data Mover can be used for a variety of functions like database security management, running SQL scripts etc. On Windows machines, you can use the Data Mover using the GUI as well as from the command prompt. While UNIX variants support only command line operation.

8. Define Records?

Fields that are grouped together as a unit are *records*.

PeopleSoft records can be defined as the collection of fields on a page. It's similar what we use tables in RDBMS.

9. What are the types of records?

- 1) - Table
- 2) - View
- 3) - Derived/Work
- 4) - Subrecord
- 5) - Dynamic View
- 6) - Query View
- 7) - Temp table

10. Define Sub records?

PeopleSoft sub records are the collection of fields that are grouped together to form an entity so that they can be used for building blocks for multiple records. Since sub records are not build through application designer they don't have their existence in the database.

11. How a Form is created?

The steps used to create a form:

- Defining the basic information for a form, including the form ID and form owner.

- Providing instructions that describe how to use the form.
- Specifying the fields that appear on the form.
- Attaching files to the form (optional).
- Defining the menu item that is used to access the form.
- Specifying the approval workflow that is required for the form.
- Previewing, testing, and activating the form.

12. Write the types of fields in forms?

- Numeric
- Text
- Date
- Time
- Yes/No
- Prompt

13. Define PAGES?

Pages are the graphical interface between your users and your application database. As a system designer, you configure or build pages that meet the data requirements of the application and that are easy to use and understand. Using PeopleSoft Application Designer, you can create, modify, and delete page definitions in your PeopleSoft system.

14. Define PeopleSoft HRMS?

Human Resources Management System (HRMS) or Human Resources Information System (HRIS), refers to the systems and processes at the intersection between human resource management (HRM) and information technology. It merges HRM as a discipline and, in particular, its basic HR activities and processes with the information technology field.

15. Define SCM?

Supply chain management (SCM) is the management of the flow of goods and services. It includes the movement and storage of raw materials, work-in-process inventory, and finished goods from point of origin to point of consumption.

16. What is CRM?

Customer relationship management (CRM) is an approach to managing a company's interaction with current and future customers. It often involves using technology to organize, automate, and synchronize sales, marketing, customer service, and technical support.

17. What are the products of Peoplesoft?

- PeopleSoft Customer Relations Management (CRM) Warehouse.
- PeopleSoft Financial Management Solutions (FMS) Warehouse.

- PeopleSoft Financials Warehouse for Public Sector and Higher Education
- PeopleSoft Human Capital Management (HCM) Warehouse.
- PeopleSoft Supply Chain Management (SCM) Warehouse.

18. What is meant by Functional PeopleSoft?

Functional People is the behavior of the data. They know how to create data and the one that is familiarize in navigating the PeopleSoft online. They give the information and specification to the Technical people also know the outcome or expected data of the program.

19. Define Financial Management Systems?

A financial management system is the methodology and software that an organization uses to oversee and govern its income, expenses, and assets with the objectives of maximizing profits and ensuring sustainability.

20. Define PeopleSoft Enterprise HRMS?

PeopleSoft HRMS is an integrated suite of applications and business processes that are based on PeopleSoft's Pure Internet Architecture (PIA) and enterprise portal technologies. The sophisticated features and collaborative, self-service functionality available in PeopleSoft HRMS.

21. Write the types of PeopleSoft Enterprise HRMS?

- PeopleSoft Enterprise HRMS integrations
- PeopleSoft Enterprise HRMS implementation

22. What is PeopleSoft Enterprise HRMS Integration?

PeopleSoft HRMS integrates with other PeopleSoft applications, such as PeopleSoft Enterprise Financials, PeopleSoft Enterprise Workforce Analytics, and PeopleSoft Enterprise Learning Management.

23. What is meant by PeopleSoft Enterprise HRMS Implementation?

PeopleSoft Setup Manager enables you to generate a list of setup tasks for your organization based on the features that you are implementing. The setup tasks include the components that you must set up, listed in the order in which you must enter data into the component tables, and links to the corresponding People Book documentation.

11 Marks

1. Describe PeopleSoft database architecture?

PeopleSoft database architecture consists of three categories as follows:

1. SYSTEM CATALOG TABLE
2. PEOPLETOOLS TABLE
3. APPLICATION TABLE

SYSTEM CATALOG TABLE

This tables are handled by DBA's since it has the DB related information along with it is used to check the performance of the PeopleSoft DB.

- Any change in the application tables updates the system catalog tables.
- Example of system catalog tables are: SYSCOLUMNS, SYSTABLES

PEOPLETOOLS TABLES

All delivered tables can be found in Peopletools Tables

- The metadata is stored in Peopletools tables like fields, records, page, component, menu, AE.
- Example of peopletools table are: PSRECDEFN, PSDBFIELD

APPLICATION TABLES

The tables we develop are stored in application tables.

- PeopleSoft has preferred the naming convention of application tables to be starting with PS_
- Example of application tables are: PS_JOB
- PS_EMPL_TBL
- PSRECDEFN
- PSDBFIELD
- PS_JOB
- PS_EMP_TBL
- SYSCOLUMNS
- SYSTABLES

Application tables are further categorized into three parts as follows:

1. Control Table
2. Transaction Table
3. Run control Table



System tables, also called system catalog tables, are analogous to a table of contents for a book or to file allocation tables on a hard drive. The structure and table names vary depending on which RDBMS you use. System catalog tables:

- Keep track of all of the objects that reside in the database instance.
- Are created by and owned by the RDBMS.

Are often described as system metadata.

People Tools tables provide the infrastructure for PeopleSoft applications by storing and managing PeopleSoft application metadata. This metadata consists of information that defines the application, such as records, fields, pages, PeopleCode, and security. People Tools tables:

- Define the structure of all object definitions that make up an application.
- Use the same table structure for all applications.
- Contain data that is added and updated only when the application is installed, or when using development tools such as PeopleSoft Application Designer or Data Mover.

Application data tables store data entered through a PeopleSoft application. The specific tables and their structures vary by application. Application data tables:

- Contain transactional data entered by users.
- Are empty prior to data entry (except the demo databases).

2. Explain in detail about Data Mover?

PeopleSoft Data Mover, is used to import / export data between databases. Apart from this, Data Mover can be used for a variety of functions like database security management, running SQL scripts etc. On Windows machines, you can use the Data Mover using the GUI as well as from the command prompt. While UNIX variants support only command line operation.

PeopleSoft Data Mover enables you to:

Transfer application data between PeopleSoft databases.

Move PeopleSoft databases across operating systems and database platforms.

Execute Structured Query Language (SQL) statements against any PeopleSoft database, regardless of the underlying operating system or database platform.

Control database security and access.

Create, edit, and run scripts.

These scripts may include any combination of SQL commands and PeopleSoft Data Mover commands for exporting and importing data.

There are two ways to run PeopleSoft Data Mover:

- **Using the Data Mover development environment.**

This is a graphical user interface (GUI), which runs only in Microsoft Windows. Use the Data Mover shortcut in the PeopleSoft program group. Select Start, Programs, *your_PSFT_program_group*, Data Mover.

- **Using the Data Mover command-line interface.**

The command-line interface is intended mainly for UNIX servers. You run PeopleSoft Data Mover from a console in Microsoft Windows and from a telnet session in UNIX. PeopleSoft Data Mover **operating modes** are:

- **Regular mode.**

Most of the time, you use regular mode. To sign in to regular mode, enter your PeopleSoft user ID and password during sign-in. In regular mode, all commands are valid.

- **Bootstrap mode.**

In bootstrap mode, you use a database access ID and password when signing in. Here use bootstrap mode for database loading, because no PeopleSoft security tables are established and also use bootstrap mode for running security commands ENCRYPT_PASSWORD.

3. Explain the types of Records in detail?

Fields that are grouped together as a unit are *records*.

PeopleSoft records can be defined as the collection of fields on a page. It's similar what we use tables in RDBMS. The difference in PeopleSoft records is the structure of PeopleSoft records is stored in metadata tables that can be defined or edited through peoplesoft application designer.

Seven types of PeopleSoft Records :

- 1) - Table
- 2) - View
- 3) - Derived/Work
- 4) - Subrecord
- 5) - Dynamic View
- 6) - Query View
- 7) - Temp table

1) **Table:** A table in PeopleSoft is similar like other database tables which has fields and that can be built and alter using application designer in PeopleSoft.

2) **View:** PeopleSoft View is a query written specially to retrieve/modify data on the records and also view's are used to show transaction data on reports and much more.

3) **Derived/Work:** Derived/work record are not built by Application Designer since there is no option to do that. It is just used for temporary storage and display purpose and the derived/work record data is saved never anywhere.

4) **Subrecord:** PeopleSoft subrecords are the collection of fields that are grouped together to form an entity so that they can be used for building blocks for multiple records. Since subrecords are not build through application designer they don't have their existence in the database.

5) **Dynamic View:** With dynamic view we can define a record definition that can be used like a view in pages and PeopleCode and is not actually stored as a SQL view in the database.

6) **Query View:** With Query view we can define the record definition as a view that is created using the PS Query. This kind of view when created by PeopleSoft Application Designer it prompts you to save the definition.

7) **Temp table:** Temporary tables in peoplesoft are there to avoid the signification risk of data contention and helps to improve performance since the application data is getting processed in temporary table not on the large application data tables.

4. Explain Multiple Windows?

IE 8 can open multiple unique sessions (but only when using the new IE 8 menu option called "New Session" or when creating a new shortcut with a special -nomerge instruction sent to IE executable) - do not use Windows Start > Internet Explorer if you want a new session - IE 7 can open as well new unique sessions but each time one should use the Windows Start > Internet Explorer to start a new browser instance (this is opposite to IE 8 new behavior - see above) - Using New Window (from any IE version or Firefox ...from menu or using CTRL+N ...or from PIA "New Window" links) will always open a new window tied to the same browser session - this will share the same cookies with previous windows.

5. Describe in detail about FORMS?

The Forms and Approval Builder enables to design online forms, specify the approval process they require, and deploy them to users within your organization. Use this feature to convert manual procedures within your organization to paperless processes that include workflow-based approvals and an audit trail for tracking progress. No coding is required on your part, and future upgrades to your PeopleSoft system will not require you to update these forms, since the forms you create are not customizations.

Once a form is published, users navigate to the form using the main menu, complete the fields and submit it for approval. Each published form includes three tabs: Form, Instructions, and Attachments. An audit trail of the approval history and comments is automatically generated as the form goes through the approval process. You can review the audit trail to see the history for each step of the approval chain.

Form Design

To create forms, use the Design Form Wizard component (FORM_DESIGN_WIZARD), which guides you through the following steps of the form creation process:

- Defining the basic information for a form, including the form ID and form owner.
- Providing instructions that describe how to use the form.
- Specifying the fields that appear on the form.
- Attaching files to the form (optional).
- Defining the menu item that is used to access the form.
- Specifying the approval workflow that is required for the form.
- Previewing, testing, and activating the form.

Forms can contain fields arranged into one or two columns. You can specify which fields are required, and define the edits that a field must pass in order for the completed form to be saved. As you design the form, you can use the following field types:

- Numeric
- Text
- Date
- Time
- Yes/No
- Prompt

Enables form users to select values from existing PeopleSoft records to complete the field.

- Code

Enables you to build a list of values that appear in a drop-down list. Form users can select a value from the list to complete the field.

- Section

Enables you to organize the form into multiple sections.

Form Status

It is important to understand the distinction between a form and a form instance. For the purposes of this documentation, when we use the term *form*, we are referring to the “master” form, or template, that form designers create using the Form Design Wizard component. A *form instance* is a deployed form that has been completed by a form user. Each form instance is automatically assigned a unique sequence number, so all completed forms can be tracked and managed independently.

The possible status values for a *form* are:

| | |
|-------------------------|---|
| <i>In Design</i> | Indicates a form that is being designed, and is not active. |
|-------------------------|---|

| | |
|------------------|--|
| Activated | <p>Indicates a form that is active and available for form users to complete.</p> <p>Depending on your user role, you can activate a form using the following pages:</p> <ul style="list-style-type: none"> • Design Form: Complete page of the Form Design Wizard component. • Manage Forms page. <p>Note: Only the owner of a form or a form administrator can activate/inactivate a form or reassign a form to a new owner.</p> |
| Inactive | <p>Indicates a form that has been inactivated by either the form's owner or a form administrator. Form administrators or form owners can inactivate forms by using the Manage Forms page. When a form is inactive, form users are not permitted to complete the form.</p> |

The possible status values for a *form instance* are:

| | |
|---------------------------------------|--|
| Initial | When a form user accesses an activated form and begins to complete it, the status of the form instance is set to <i>Initial</i> . |
| Pending | When the form user finalizes the form instance and submits it for approval, the status changes to <i>Pending</i> . |
| On Hold | When an approver or reviewer requests more information about a submitted form, the status changes to <i>On Hold</i> . |
| Approved, Denied, or Cancelled | As the form instance flows through the required approval chain, the status subsequently updates to approved, denied, or cancelled. |

Security and Delivered Roles

User roles determine who has permission to access, design, and administer forms. The following table lists the delivered roles and associated permission lists for Form and Approval Builder.

| Role Name | Description | Permission Lists | Access Rights |
|---------------|---|----------------------|--|
| FORM_USER | Form user: Can complete and submit forms. | EOFM1000 | The form user has add/update/display access to forms that have been created with the Form Design Wizard (form instances). |
| FORM_DESIGNER | Form designer: Can complete and submit forms, design forms, and manage forms that they are assigned to as the owner. | EOFM1000 EOFM2000 | The form designer has add/update/display access to forms that have been created with the Form Design Wizard, <i>and</i> has add/update/display access to |

| Role Name | Description | Permission Lists | Access Rights |
|--------------------|---|--|--|
| | | | the Form Design Wizard component. Form designers have update/display access to only <i>their</i> forms when using the Manage Forms (FORM_RPT) component. |
| FORM_ADMIN | Form administrator: Can complete and submit forms, design forms, and manage <i>all</i> forms. | EOFM1000 EOFM2000 EOFM3000 EOSD2000 | The form administrator has the same access as the form designer, and additionally, has update/display access to <i>all</i> forms when using the Manage Forms (FORM_RPT) component. |
| FORM_CI_DEVELOPERS | Form to component interface (CI) developer: Can complete and submit forms, design forms, manage <i>all</i> forms, and map forms to component interfaces (integrate forms). | EOFM1000 EOFM2000 EOFM3000 EOFM4000 EOSD2000 | The form to CI developer has the same access as the form administrator, and in addition can set up form integration by mapping forms to component interfaces using the Forms to CI Mapping page. |

6. Explain PAGES in detail?

Pages are the graphical interface between the users and the application database. As a system designer, we configure or build pages that meet the data requirements of the application and that are easy to use and understand. Using PeopleSoft Application Designer, we can create, modify, and delete page definitions in your PeopleSoft system.

The page design based on the type of data that we plan to access and maintain. In some cases, a page references a single record definition; in others, you may want to reference multiple records. To accommodate a variety of page designs, PeopleSoft Application Designer uses *level-based controls*. The three level-based controls are:

- Grids
- Scroll areas
- Scroll bars

You specify the level of these controls by setting the *occurs levels* on the Record tab of the page field properties dialog box for the level-based control that you are configuring. Level-based controls have four levels:

- Level 0

This level directly corresponds to the high-level key information of the underlying record. Level 0 fields are usually physically located at the top of the page, set to be display-only, and display the data that the user entered on the initial search page.

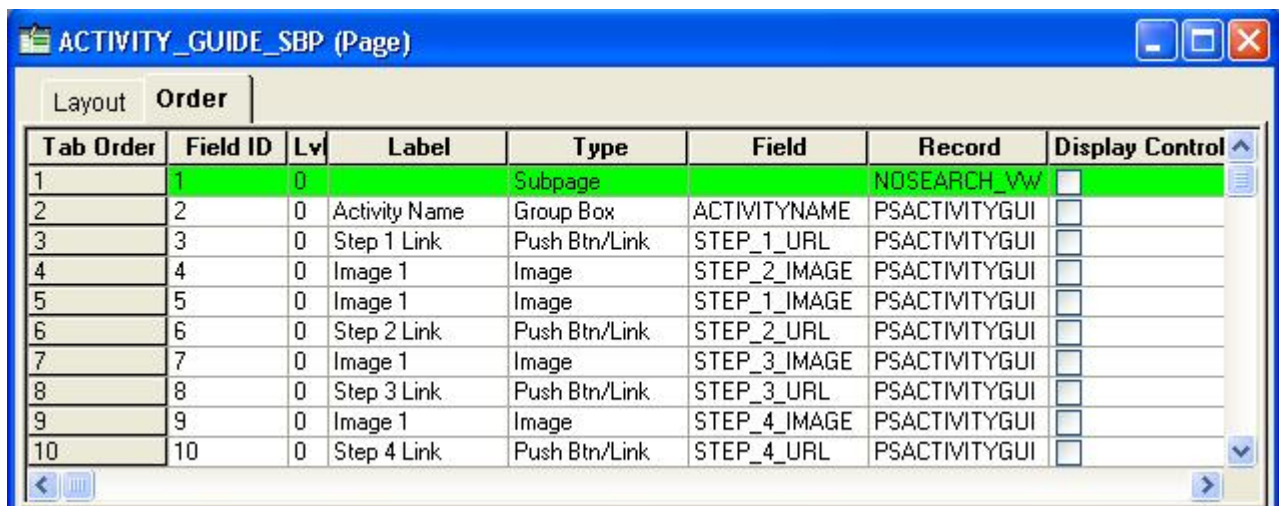
- Levels 1–3

These levels correspond to lower level key information that is subordinate to the Level 0 key values. Level 1 is subordinate to Level 0; Level 2 is subordinate to and nested within Level 1; and Level 3 is subordinate to and nested within Level 2.

The first occurs level on a page is Level 0. In general, this level is reserved for the primary key fields that are used to search for data. A page might not contain any level-based controls; in such a case, all fields are set to Level 0. This case is particularly true for secondary pages or subpages that contain few data entry fields, as shown in the following example. The third column, Lvl (Level), that appears in the grid on the Order tab indicates that all fields on the page are at Level 0:

Image: Example of page definition Order tab

This denotes the fields and controls on the Example of page definition Order tab.



| Tab Order | Field ID | Lvl | Label | Type | Field | Record | Display Control |
|-----------|----------|-----|---------------|---------------|--------------|---------------|--------------------------|
| 1 | 1 | 0 | | Subpage | | NOSEARCH_VW | <input type="checkbox"/> |
| 2 | 2 | 0 | Activity Name | Group Box | ACTIVITYNAME | PSACTIVITYGUI | <input type="checkbox"/> |
| 3 | 3 | 0 | Step 1 Link | Push Btn/Link | STEP_1_URL | PSACTIVITYGUI | <input type="checkbox"/> |
| 4 | 4 | 0 | Image 1 | Image | STEP_2_IMAGE | PSACTIVITYGUI | <input type="checkbox"/> |
| 5 | 5 | 0 | Image 1 | Image | STEP_1_IMAGE | PSACTIVITYGUI | <input type="checkbox"/> |
| 6 | 6 | 0 | Step 2 Link | Push Btn/Link | STEP_2_URL | PSACTIVITYGUI | <input type="checkbox"/> |
| 7 | 7 | 0 | Image 1 | Image | STEP_3_IMAGE | PSACTIVITYGUI | <input type="checkbox"/> |
| 8 | 8 | 0 | Step 3 Link | Push Btn/Link | STEP_3_URL | PSACTIVITYGUI | <input type="checkbox"/> |
| 9 | 9 | 0 | Image 1 | Image | STEP_4_IMAGE | PSACTIVITYGUI | <input type="checkbox"/> |
| 10 | 10 | 0 | Step 4 Link | Push Btn/Link | STEP_4_URL | PSACTIVITYGUI | <input type="checkbox"/> |

When you add a level-based control, a scroll area or a grid, to your page, the default occurs level for that control is set to 1 in the properties dialog box. If you place a field in or below that scroll area, it is also set to Level 1 on the Order tab, even if it is another level-based control. If necessary, you can use the Set to Level 0 feature of the horizontal rule control to restart the occurs level on a page.

Nesting controls involve two or more level-based controls on a page, such as two scroll areas, when the second scroll area has an occurs level of 2. You nest controls when the new data that you want to add is a repeating set of data for each entry in your first level-based control. In doing so, you create a hierarchical, or parent and child, relationship between the controls and the processing of the record definitions. The Level 2 control is the child of and is subordinate to the Level 1 control.

For example, on the Compensation page, the Level 2 grid is nested in the Level 1 scroll area. On the Amounts tab of the grid, each compensation action in the Level 1 scroll area has two rate codes:

Image: Example of Compensation page with nested grid in scroll area

This example illustrates the fields and controls on the Example of Compensation page with nested grid in scroll area.

Compensation Find First 1 of 1 Last

Effective Date: 01/01/2000 Go To Row

Effective Sequence: 0 Action: Hire

HR Status: Active Reason:

Payroll Status: Active Job Indicator: Primary Job Current

Compensation Rate: 1,776.67 AUD *Frequency: M Monthly

► Comparative Information

► Pay Rates

Default Pay Components Contract Change Prorate Option

Pay Components Customize Find First 1-2 of 2 Last

Amounts Controls Changes Conversion

| | *Rate Code | Seq | Comp Rate | Currency | Frequency | Points | Percent | Rate Code Group |
|---|------------|-----|--------------|----------|-----------|--------|---------|-----------------|
| 1 | KABS01 | 0 | 1,776.667000 | AUD | M | | | |
| 2 | KABSTL | 0 | 1,950.500000 | AUD | M | | | |

Calculate Compensation

This example of the Additional Pay 1 page shows three levels of data, two of which are nesting scroll areas in the Earnings scroll area:

Image: Example of Additional Pay 1 page with nested scroll areas

This example illustrates the fields and controls on the Example of Additional Pay 1 page with nested scroll areas. You can find definitions for the fields and controls later on this page.

Additional Pay 1 | **Additional Pay 2** | Additional Pay 3

Albion, Charles Employee ID: Z1000 Empl Rcd#: 0

Earnings Find | View All First 1 of 1 Last

*Code:

Eff Date and Default Job Data Find | View All First 1 of 1 Last

Effective Date: 07/05/2000

Compensation Rate/Frequency: \$6,000.000000 Monthly

Standard Hours: 40.00 Employee Type: Salaried

Or Overrides to Job Data Find | View All First 1 of 1 Last

*Addl Seq #: Earnings End Date:

Rate Code: Reason: Not Specif

Earnings: Hours: Hourly Rate:

Goal Amt: Goal Bal:

Sep Chk #: ☐ ☐ Disable Direct Deposit ☐ Prorate Additional Pay ☐ OK to Pay

Applies to Pay Periods: ☒ First ☐ Second ☐ Third ☐ Fourth ☐ Fifth

Save Return to Search Next in List Previous in List Update/Display Include History Correct History

[Additional Pay 1](#) | [Additional Pay 2](#) | [Additional Pay 3](#)

In this case, we want to maintain information about the compensation history for a single employee and, for each change or override in compensation status, enter details about that change. The first scroll area, Earnings, associates the earning code with the second, subordinate scroll area, Eff Date (effective date) and Default Job Data, which enables the user to enter the effective date for the status change. The third scroll area, Or Overrides to Job Data, which is subordinate to the second, enables the user to enter the overrides to the default job data that appears in the second scroll area.

For each of the three scroll areas, you can have several rows of data. While you do not see actual scroll bars on your page, the navigation buttons and links in the navigation bars for each scroll area enable you to navigate or scroll through the rows.

Nested Grids in Scrolls

Here we can nest a grid in a scroll or scroll area. The following example shows the Other Earnings grid at Level 3, which is nested in the Earnings scroll area at Level 2, which is nested in the Pay sheet Details scroll area at Level 1. A nested grid serves the same function as a nested scroll area. It can offer a more compact way of viewing many fields of data, particularly if you use a tabbed grid.

Note: You can nest a grid in a scroll area, but you cannot nest a scroll area in a grid or a grid in another grid.

Image: Example of Paysheet page with three levels of nesting

This example illustrates the fields and controls on the Example of Paysheet page with three levels of nesting. You can find definitions for the fields and controls later on this page.

Paysheet | One-Time Deductions | One-Time Garnishment | One-Time Taxes | Page Balances

Company: BN1 | Pay Group: B2M | Pay Period End Date: 12/31/1998 | Page: 1

Paysheet Details | Find | First | 1 of 3 | Last

Line: 1 | Empl ID: B-R5MASET3 | Retmaset3.name
 Empl Record: 0 | Benefit Record: 0 | ☐ Manual Check
 Check Number: | Check Date: | Total Gross: | Net Pay:
 Transaction Message: No Message

Earnings | Find | View All | First | 1 of 1 | Last

☒ OK to Pay | ☒ Job Pay | ☐ Override Hourly Rate | ☐ No Direct Deposit
☐ Gross-Up | ☐ TL Records

Reg Hours: 173.33 | OT Hours: | Hourly Rate: 17.307692 | Reg Salary:
 Reg Rate Code: | OT Rate Code:
 Shift: N/A | State: CA | Locality:
 Earnings Begin: 12/01/1998 | Earnings End: 12/31/1998 | [Additional Data](#)

Other Earnings | Customize | Find | View All | First | 1 of 1 | Last

| *Code | Seq Nbr | Rate Code | Hours | Rate | Amount | Source |
|-------|---------|-----------|-------|------|--------|--------|
| | | | | | | |

Levels and Runtime Processing

Levels play an important role in runtime processing. The component processor relies on the level at which you place a field on a page to determine how to process any PeopleCode attached to the field in the record definition.

Effective Dates and Level-Based Controls

The EFFDT (effective date) field must be the *only* key field that is controlled by level-based controls that you create to help users maintain multiple occurrences of data that is keyed by effective date. Otherwise, the effective date processing for update actions does not function correctly.

The search record that you define for a component determines the key list; that is, search keys for which users are prompted when they select a page and an action. A search record can be either a view that concatenates information stored in several tables or the underlying table itself. Select the search record that contains all of the key items for the primary record underlying the page.

The keys for which the search dialog box prompts should populate the high-level (Level 0 in the page definition) key controls on a page. These key controls always appear before any level-based control on the page and are typically display-only. A search record might differ from a primary record definition, but it must contain all of the Level 0 keys that you placed on the page.

Using Keys for Views

When you create a view that should use key values from a page, each key must be:

- Listed as an output column from the view (not just referenced in the WHERE clause of the view).
- Marked as a key field of the view.

For example, suppose that a page has PERSON.PERSONID as a Level 0 field. PERSONID is the key of the PERSON record. A grid on the page uses the PERSON_ADDRESS_VW view to display addresses for the current person. The SQL statement for the view cannot only mention PERSON.PERSONID in the WHERE clause; PERSONID must be both an output column of the view and a key field of the view.

On some pages, you may want a few of the field controls to display multiple rows or occurrences of data. To do this, you can add a level-based control: a scroll area, a grid, or a scroll bar. Users can then add, edit, delete, find, and scroll through multiple occurrences of data in a page control or group of controls using action buttons, links, or the browser scroll bar, depending on how you set the occurs count. Using a scroll area or a grid, rather than a scroll bar, is the preferred page design to show multiple occurrences of data.

For example, this Checklist Table page contains the Assignment Checklist Item grid with an occurs count of 5. The Checklist Item scroll area in which it is nested must have an occurs count of 1 since we can see only one row of data in the scroll area:

Image: Example of Person Checklist page with a grid of occurs count 5 nested in a scroll area of occurs count 1

This example illustrates the fields and controls on the Example of Person Checklist page with a grid of occurs count 5 nested in a scroll area of occurs count 1. You can find definitions for the fields and controls later on this page.

Person Checklist

Susan Jones Person ID: 0002

Checklist History Find | View All First 1 of 1 Last

*Checklist Date: 08/17/2011 31

*Checklist: DEUTER Termination

Comment:

| *Sequence | *Item Code | Description | *Status | Link ID |
|-----------|------------|------------------------------|-----------|--|
| 100 | TER01 | Cancel Security Access | Initiated | |
| 200 | TER02 | Update Social Insurance Data | Initiated | Update Social Insurance Data |
| 600 | TER06 | Holiday status/ bonus | Initiated | |
| 800 | TER08 | Process Job Change | Initiated | Job Change Request |
| 900 | TER09 | Verify Termination Date | Initiated | Verify Termination Date |

Customize | Find | First 1-5 of 10 Last

To see the next set of five rows in the Assignment Checklist Item grid, a user can click the right arrow button. To see all rows of data at once, in either the grid or scroll area, the user can select the View All link in the navigation bar.

In PeopleSoft Application Designer, you set the occurs count for a level-based control on the General tab in the properties dialog box. You can set the occurs count to any number. You also have the option of selecting Unlimited Occurs Count so that the user sees all rows of data. If the page has more options than can be viewed in the window at runtime, the browser displays a scroll bar enabling the user to scroll to the bottom of the page.

You can enable your users to look up the valid values that they can enter in a field. For this, PeopleSoft Application Designer provides prompts, or lookup buttons. Your PeopleSoft applications use the following three types of prompts:

- Drop-down list box

This type of prompt generates a list of values, which appears directly below the prompt field when the user clicks the down arrow that appears beside the field.

The drop-down list box is a predefined control that you place on your page. The Job Indicator field on the Work Location page is an example of drop-down list box.

- Calendar prompt

This type of prompt generates a small calendar, which appears next to the date field, when the user clicks the calendar button. Users can navigate the calendar to find and select a date. The Effective Date, Position Entry Date, and Department Entry Date fields on the Work Location page are examples of calendar prompts. The calendar prompt appears automatically as part of any edit box control that you associate with a date field as long as the Show Prompt Button check box display option on the Record tab is selected.

- Lookup button

A prompt or lookup button opens a lookup page in the user's browser populated with up to 300 available values for that field. The user can then either select the desired value or refine their search further. For extremely large tables, the system administrator has the option of excluding that table from auto prompting on the Lookup Exclusion Table via PeopleTools, Utilities, Administration.

In PeopleSoft Application Designer, you can only associate a lookup button with an edit box control. The look up button appears automatically as part of any edit box control as long as the Show Prompt Button check box display option on the Record tab is selected and the record field with which you associate the edit box lists values in either the Translate table or a prompt table.

Image: Work Location page showing sample prompt fields

This example illustrates the fields and controls on the Work Location page showing sample prompt fields.

Work Location Job Information Job Labor Payroll Salary Plan Compensation

Jaime Salazar
Employee

Empl ID: KYLA15
Empl Record: 0

Find First 1 of 1 Last

Go To Row + -

*Effective Date: 01/01/2008
Effective Sequence: 0
HR Status: Active
Payroll Status: Active

Position Number:
Override

Position Entry Date:
Position

*Regulatory Region: MEX
*Company: KYF
*Business Unit: KY001
*Department: 22000
Department Entry Date: 01/01/2000
*Location: KYDF
Establishment ID:
Date Created: 12/18/2008

Last Start Date: 01/01/2000
Expected Job End Date:
Spain
Japan

Job Data Employment Data Earnings Distribution Benefits Program Participation

Hidden pages are work pages that are associated with derived or work records; they are often used in work groups. You can store all of your work field controls on hidden pages. Create these pages when you want PeopleCode calculations to be performed in the background so that the user does not see them. As a convention, the names of work pages that are delivered with your application names end with the suffix *_WRK*; you should follow the same naming convention. An example name is: *MC_TYPE_WRK*.

Hide a page by selecting the Hidden check box in the component grid as you set up the component definition.

7. Explain the Peoplesoft HRMS Database?

THE PEOPLESFT HRMS DATABASE:

All PeopleSoft products attempt a balance between functional requirements (what works best for humans) and relational database requirements (what works best for computers). This issue is especially relevant to human resources requirements. Human resources professionals must be customer-oriented, and this requires extraordinary flexibility. Casual statements such as "We have to cut 5,000 checks by tomorrow morning!" or "We just bought XYZ corporation and need to add 30,000 employees to our system!" can bring an entire human resources department to tears. PeopleSoft's role is to simplify these seemingly drastic changes.

Functional flexibility, the ability to perform human resources feats in a single click, is a cornerstone of the PeopleSoft product. On the other hand, PeopleSoft is a software product with

many programmers behind it. These programmers work together, writing separate components and improving technical benchmarks, like access time and storage capabilities. The companies who purchase PeopleSoft also typically employ programmers who modify and maintain the product. The “user,” then, could be a programmer or a VP of human resources. The computer and human effort required to maintain an infinitely flexible system becomes overwhelming.

To prevent a tangled web of data, PeopleSoft stores data using a database organizational concept called “data integrity.” The philosophy of data integrity is simple: a database stores data efficiently by storing it once, and only once. Duplication equals inefficiency and creates a chance that data will be changed in one place and not another. The efficient storage of data assures its integrity. It is a mantra followed by database programmers, and PeopleSoft is no exception. Data integrity keeps the database happy and allows it to grow as functionality is added to PeopleSoft. Yet it divides the human resources data we need into over 5,611 tables. With each new PeopleSoft release, data is spread thinner and thinner, to more and more tables.

For example, PeopleSoft took the Social Security Number and made three new tables to store it. Now PeopleSoft can store different forms of the SSN for different countries with different validations. As the functional flexibility of the program grows, so does the size of the database.

8. What is PRODUCTS. Explain it in detail.

PeopleSoft Accelerator

A small change can have a disastrous effect on your PeopleSoft system. Every update or change requires testing to make sure it doesn't bring your business down. All that testing takes time and money away from your business. For businesses that rely on HP Software Quality Management and PeopleSoft, we've helped you get started with a platform and have prebuilt the business-process tests for you. Validate your critical cross-platform business processes seamlessly, reducing risk while being more responsive.

A Full Automation Framework

Our Accelerator Platform provides a complete environment with a structured approach ready for fast customization. By the time implementation is over, you'll have a library of test cases that cover a vast array of business processes customized for your unique environment.

Scriptless Testing For:

- Patches
- Hot Packs
- Upgrades
- Technology Stack Upgrades or Modifications

Leverage Your Existing HP Software Quality Management Solution

There's no need to rip and replace to get the benefits of scriptless testing. Accelerator works with HP Software's Quality Management products, including Quality Center, QuickTest Professional and Business Process Testing.

Generate Thorough Documentation

No guesswork here. Accelerator documents all tests performed and the results achieved with the meticulous accuracy that manual testers only experience in their nightmares. Audit and compliance made easy - reporting can now be done with no additional costs or complexity. We also throw in a full visual test history for auditors and extended reporting.

9. Explain Peoplesoft HRMS Functional in detail?

Step 1: Basic Setup HRMS

Create SETID

Navigation:

Main Menu ->PeopleTools ->Utilities ->Administration ->TablesetID's

Create Business Unit

Navigation:

Main Menu ->Setup HRMS ->Foundation Tables ->Organisation ->Business Unit

Setup Location:

Navigation:

Main Menu ->Setup HRMS ->Foundation Tables ->Organisation ->Location

Add a company

Navigation:

Main Menu ->Setup HRMS ->Foundation Tables ->Organisation ->Company

Create an Establishment ID

Navigation:

Main Menu ->Setup HRMS ->Foundation Tables ->Organisation ->Establishment

Create Departments

Navigation:

Main Menu ->Setup HRMS ->Foundation Tables ->Organisation ->Departments

Setting the Business Unit options default

Navigation:

Main Menu ->Setup HRMS ->Foundation Tables ->Organisation ->Business Unit options default
(Enter the company and Country details)

Create a pay Group Table:

Navigation:

Main Menu ->Setup HRMS ->Product Related ->Pay Roll ->Pay Group Table

Create a Job Code

Navigation:

Main Menu ->Setup HRMS ->Foundation Tables ->Job Attributes ->Job Code Table

Setting Org Defaults permission lists

Navigation:

Main Menu ->Setup HRMS ->Foundation Tables ->Organisation ->Org Defaults by permission
Lists (Enter the Organisation related
data here)

Tree Manager:

Navigation:

Main Menu ->Tree Manager->Tree Utilites ->Tree Manager

->Create a new tree with the name DEPT_SECURITY

->Select the structure ID as DEPARTMENT

->Select the category as HR

->Click on ADD LEVEL and add the following below levels

1.Corporate

2.Business

3.Region

4.Company

5.Department

6.Division

7.Group

8.Unit

->Select the root node

->save it.

Note: In Business Options Default enter the details.

Security:

Navigation:

Main Menu ->Security ->Core Row Level Security ->Security by Permission Lists
->Select HCDPALL.Add ur business unit it and select the 011 as Security access type.

Security by Dept Tree:

Navigation:

Setup HRMS ->Security ->Core Row level Security ->Security by Dept Tree
->Select HCDPALL.Add your SETID and Department ID in it.

Running the Process:

Navigation:

Setup HRMS ->Security ->Core Row Level Security->Refresh SJT Class ALL / Refresh Trans SJT
Tables / Refresh SJT OPR class

Run all the above process

Buzz the Application

Navigation:

Setup HRMS ->Install -> Installation Table
->Change the SETID and Campany Name.
->Restart the services.

Adding a Person:

Navigation:

Main Menu ->Workforce Administration -> Personal Information ->Add a Person.

HRMS Functional Training Course Content

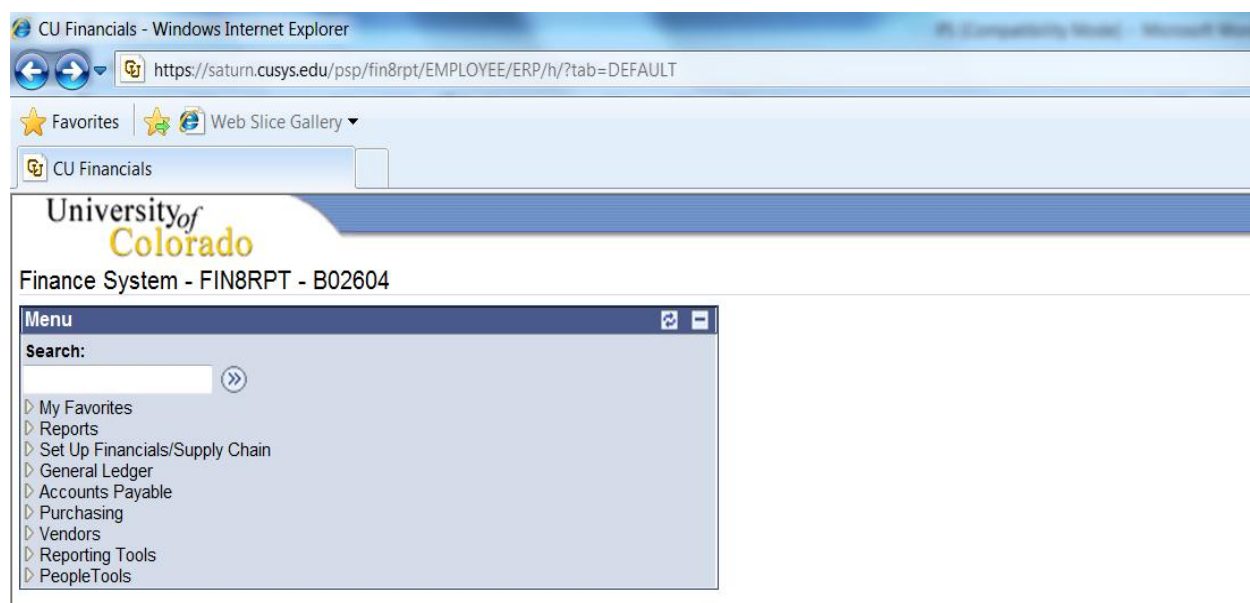
- Introduction to ERP
- Introduction different types of PeopleSoft Job roles
- Information on Modules in PeopleSoft.
- Navigation on PeopleSoft system (Introduction to App Designer)
- Introduction to all the navigation in the application.
- Setup HRMS

- Position Management
- Workforce Admin
- ESS and MSS
- Base Benefits
- Security
- Tree Manager
- Reporting Tools

10. Explain in detail about Financial Management System?

The Finance System (PeopleSoft) offers a wide range of reports and queries. To access the PeopleSoft reporting system, you will need to log on to MyCUInfo. The link to the PeopleSoft application is under the “CU Resource” tab of the MyCUInfo webpage.

Once you get into PeopleSoft, the following menu will appear:



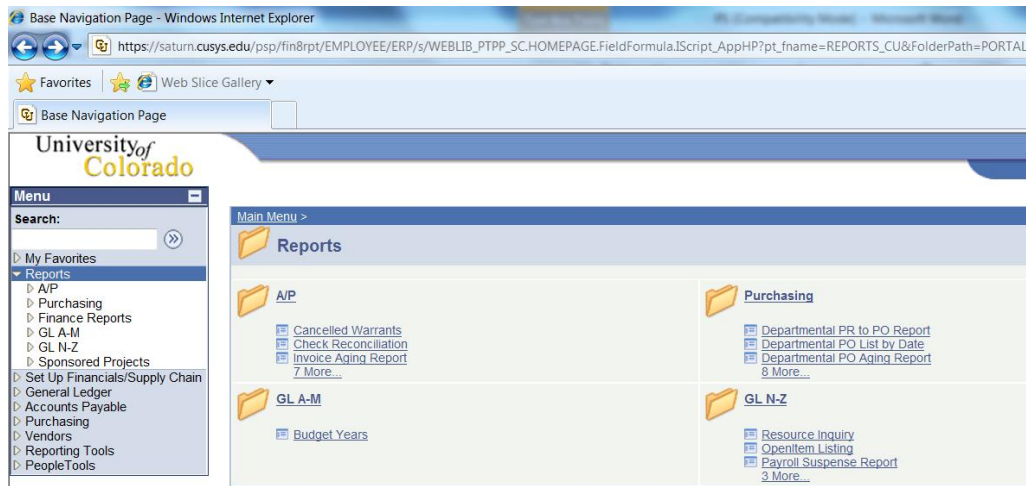
Most financial reports (balance sheet, summary of transactions, etc.) will be under the “Reports” section. The “General Ledger” section provides lookup and query tools. The “Reporting Tool” section is where your reports will be posted. The following 4 sections provide step by step procedures to run balance sheet report, summary of transactions report, detail of transactions report, and journal entry query.

1. Balance Sheet:

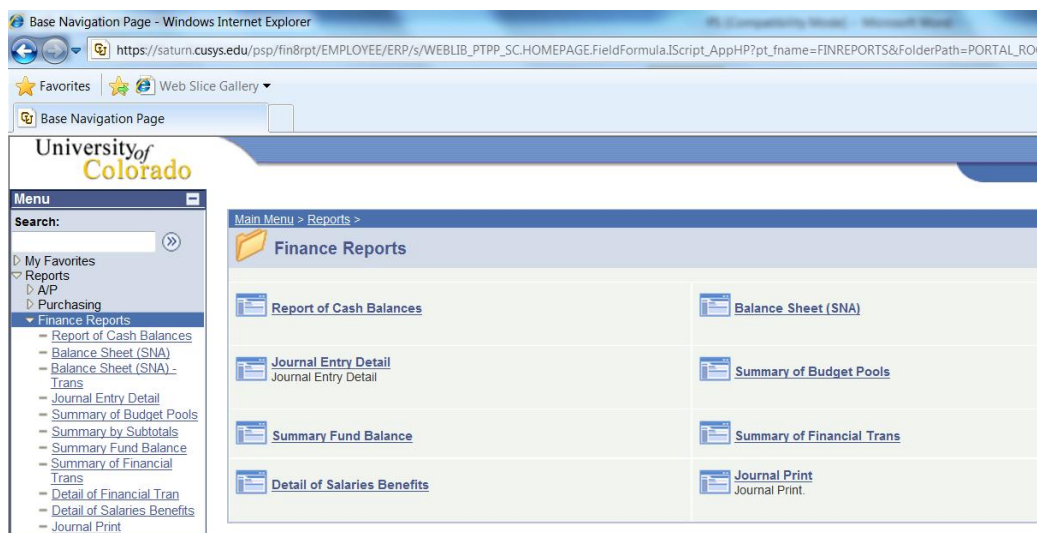
PeopleSoft can run balance sheet reports by project, program, speedtype and org.

1. Then select “Finance Reports”
2. Click on the “Reports” menu to expand it.

Select “Balance Sheet”

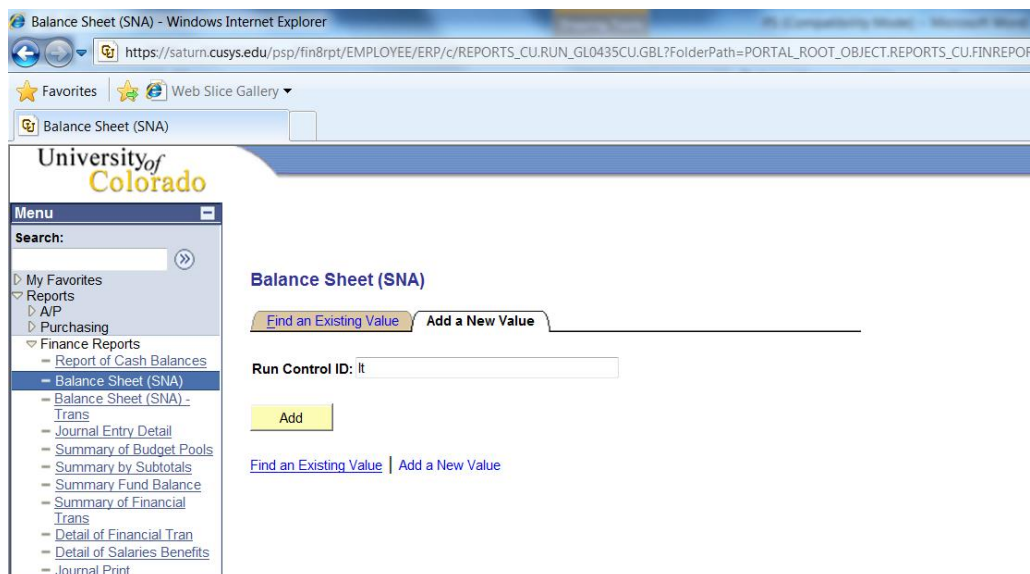


Select "Balance Sheet"



2. Enter a run control ID of your choice and click the "Add" button

1. Select the "Add a New Value" tab



1. Enter the page break code. Click on the magnifying glass icon to guide you as needed

2. Enter Fiscal Year and period
3. Click on the “Run” button
4. Enter Org, program, project, or speetype

The screenshot shows the University of Colorado Financial Reporting System interface. On the left is a 'Menu' sidebar with a search bar and a tree view of reports. The 'Balance Sheet (SNA)' report is selected. The main area displays the 'Run GI0435cu' dialog box. It includes fields for 'Fiscal Year' (2011), 'Period' (10), 'Fund Code', 'ORG Tree Node', 'Org', 'Program Code', 'Project', and 'Report Version' (2011). A 'SpeedType' dropdown is also present. A 'Please Note' message states: 'You must specify legal size paper on your printer setup to print this report'. There are buttons for 'Run Control ID: It', 'Report Manager', 'Process Monitor', 'Run', 'Save', 'Add', and 'Update/Display'.

2. Then click on the “Reporting Tool” menu to expand it
1. Click the “OK” button

Select “

The screenshot shows the University of Colorado Financial Reporting System interface. On the left is the 'Menu' sidebar. The main area displays the 'Process Scheduler Request' dialog box. It includes fields for 'User ID' (B02604), 'Run Control ID: It', 'Server Name', 'Run Date' (04/06/2011), 'Recurrence', and 'Run Time' (4:37:51 PM). There is a 'Reset to Current Date/Time' button. Below these fields is a 'Process List' table with columns: 'Select', 'Description', 'Process Name', 'Process Type', '*Type', '*Format', and 'Distribution'. The table contains one entry: 'Balance Sheet (SNA)' with process name 'GL0435CU' and process type 'SQR Report'. At the bottom are 'OK' and 'Cancel' buttons.

Select “Report Manager”

The screenshot shows the University of Colorado Financial Reporting System interface. On the left is the 'Menu' sidebar. The main area displays the 'Reporting Tools' menu. It includes a 'Main Menu >' button and a 'Reporting Tools' section with the text 'Run, create, and manage queries and nVision reports.' Below this are two buttons: 'Report Manager' (with a subtext 'Review report list.') and 'Process Monitor' (with a subtext 'Review process and').

2. When your report becomes available, a hyperlink (in blue underlined font) will appear. Click on it to view your report.

1. Your report will be added to the report list below, but it usually is not available right away. Refresh this page as needed to get an update on the status of your report.

University of Colorado

Menu

Search:

- My Favorites
- Reports
- Set Up Financials/Supply Chain
- General Ledger
- Accounts Payable
- Purchasing
- Vendors
- Reporting Tools
 - Report Manager
 - Process Monitor
- People Tools

Administration Archives

View Reports For

Refresh Last: 1 Days User ID: B02604 Type: Folder:

Instance: to: Status:

Report List

| Select | Report ID | Prcs Instance | Description | Request Date/Time | Format | Status | Details |
|--------------------------|-----------|---------------|---------------------|----------------------|-----------------|--------|-------------------------|
| <input type="checkbox"/> | 1377417 | 4354311 | Balance Sheet (SNA) | 04/06/2011 4:39:20PM | Acrobat (*.pdf) | Posted | Details |

☒ Select All ☐ Deselect All

Delete Click the delete button to delete the selected report(s)

2. Summary of Financial Transaction:

From the “Financial Reports” menu, select “Summary of Financial Trans”

University of Colorado

Menu

Search:

- My Favorites
- Reports
 - A/P
 - Purchasing
 - Finance Reports
 - Report of Cash Balances
 - Balance Sheet (SNA)
 - Balance Sheet (SNA) - Trans
 - Journal Entry Detail
 - Summary of Budget Pools
 - Summary by Subtotals
 - Summary Fund Balance
 - Summary of Financial Trans
 - Detail of Financial Tran
 - Detail of Salaries Benefits
 - Journal Print

Main Menu > Reports >

Finance Reports

- Report of Cash Balances
- Balance Sheet (SNA)
- Journal Entry Detail
- Summary of Budget Pools
- Summary Fund Balance
- Summary of Financial Trans
- Detail of Salaries Benefits
- Journal Print

2. Enter a run control ID of your choice and click the “Add” button

1. Select the “Add a New Value” tab

University of Colorado

Menu

Search:

- My Favorites
- Reports
 - A/P
 - Purchasing
 - Finance Reports
 - Report of Cash Balances
 - Balance Sheet (SNA)
 - Balance Sheet (SNA) - Trans
 - Journal Entry Detail
 - Summary of Budget Pools
 - Summary by Subtotals
 - Summary Fund Balance
 - Summary of Financial Trans**
 - Detail of Financial Tran
 - Detail of Salaries Benefits
 - Journal Print

Summary of Financial Trans

Find an Existing Value | Add a New Value

Run Control ID:

Add

Find an Existing Value | Add a New Value

1. Enter Fiscal Year and period
2. Enter Org, program, project, or speetype
3. Enter the page break code. Click on the magnifying glass icon to guide you as needed
4. Click on the "Run" button

University of Colorado

Menu

Search:

- My Favorites
- Reports
 - A/P
 - Purchasing
 - Finance Reports
 - Report of Cash Balances
 - Balance Sheet (SNA)
 - Balance Sheet (SNA) - Trans
 - Journal Entry Detail
 - Summary of Budget Pools
 - Summary by Subtotals
 - Summary Fund Balance
 - Summary of Financial Trans**
 - Detail of Financial Tran
 - Detail of Salaries Benefits
 - Journal Print
 - GL A-M
 - GL N-Z
 - Sponsored Projects
 - Set Up Financials/Supply Chain
 - General Ledger
 - Accounts Payable
 - Purchasing
 - Vendors
 - Reporting Tools
 - PeopleTools

Summary

Run Control ID:

Report Manager | Process Monitor | Run

Report Request Parameters

Fiscal Year: Speed Type:

Period:

Fund Code:

ORG Tree Node:

Org:

Program:

Project:

Sub-class:

Page Break By:

Export to Excel:

Save

Add | Update/Display

2. Then click on the "Reporting Tool" menu to expand it
1. Click the "OK" button

University of Colorado

Menu

Search: []

- My Favorites
- Reports
 - A/P
 - Purchasing
 - Finance Reports
 - Report of Cash Balances
 - Balance Sheet (SNA)
 - Balance Sheet (SNA) - Trans
 - Journal Entry Detail
 - Summary of Budget Pools
 - Summary by Subtotals
 - Summary Fund Balance
 - Summary of Financial Trans
 - Detail of Financial Tran
 - Detail of Salaries Benefits
 - Journal Print
 - GL A-M
 - GL N-Z
 - Sponsored Projects
 - Set Up Financials/Supply Chain
 - General Ledger
 - Accounts Payable
 - Purchasing
 - Vendors
 - Reporting Tools
 - PeopleTools

Process Scheduler Request

User ID: B02604 Run Control ID: tl

Server Name: PSUNX Run Date: 04/07/2011

Recurrence: Run Time: 10:19:40AM [Reset to Current Date/Time](#)

Time Zone: []

Process List

| Select | Description | Process Name | Process Type | *Type | *Format | Distribution |
|-------------------------------------|------------------|--------------|--------------|-------|---------|------------------------------|
| <input checked="" type="checkbox"/> | Summary of Trans | GL0100CU | SQR Report | Web | PDF | Distribution |

[OK](#) [Cancel](#)

Select "Report Manager"

University of Colorado

Menu

Search: []

- My Favorites
- Reports
- Set Up Financials/Supply Chain
- General Ledger
- Accounts Payable
- Purchasing
- Vendors
- Reporting Tools
 - Report Manager
 - Process Monitor
- PeopleTools

Main Menu >

Reporting Tools

Run, create, and manage queries and nVision reports.

[Report Manager](#)
Review report list.

[Process Monitor](#)
Review process and...

1. Your report will be added to the report list below, but it usually is not available right away. Refresh this page as needed to get an update on the status of your report.
2. When your report becomes available, a hyperlink (in blue underlined font) will appear. Click on it to view your report.

University of Colorado

Menu

Search: []

- My Favorites
- Reports
 - Set Up Financials/Supply Chain
 - General Ledger
 - Accounts Payable
 - Purchasing
 - Vendors
- Reporting Tools
 - Report Manager
 - Process Monitor
- PeopleTools

Administration | Archives

View Reports For

Refresh Last: 1 Days User ID: B02604 Type: []

Instance: [] to: [] Status: [] Folder: []

Report List Customize | Find | View All | First 1-2 of 2 Last

| Select | Report ID | Prce Instance | Description | Request Date/Time | Format | Status | Details |
|--------------------------|-----------|---------------|-------------------------------------|----------------------|-----------------|--------|-------------------------|
| <input type="checkbox"/> | 1377574 | 4355197 | Summary of Trans | 04/07/2011 8:56:50AM | Acrobat (*.pdf) | Posted | Details |
| <input type="checkbox"/> | 1377417 | 4354311 | Balance Sheet (SNA) | 04/06/2011 4:39:20PM | Acrobat (*.pdf) | Posted | Details |

☒ [Select All](#) ☐ [Deselect All](#)

[Delete](#) Click the delete button to delete the selected report(s)

[Save](#)

Administration | Archives

1. Detail of Financial Transactions:

From the “Financial Reports” menu, select “Detail of Financial Trans.” Follow the same steps as detailed in the “Summary of Transactions” above to complete this report.

University of Colorado

Menu

Search: []

- My Favorites
- Reports
 - A/P
 - Purchasing
 - Finance Reports
 - Report of Cash Balances
 - Balance Sheet (SNA)
 - Balance Sheet (SNA) - Trans
 - Journal Entry Detail
 - Summary of Budget Pools
 - Summary by Subtotals
 - Summary Fund Balance
 - Summary of Financial Trans
 - Detail of Financial Tran
 - Detail of Salaries Benefits
 - Journal Print

Main Menu > Reports >

Finance Reports

- [Report of Cash Balances](#)
- [Balance Sheet \(SNA\)](#)
- [Journal Entry Detail](#)
Journal Entry Detail
- [Summary of Budget Pools](#)
- [Summary Fund Balance](#)
- [Summary of Financial Trans](#)
- [Detail of Salaries Benefits](#)
- [Journal Print](#)
Journal Print.

2. Journal Entry Query:

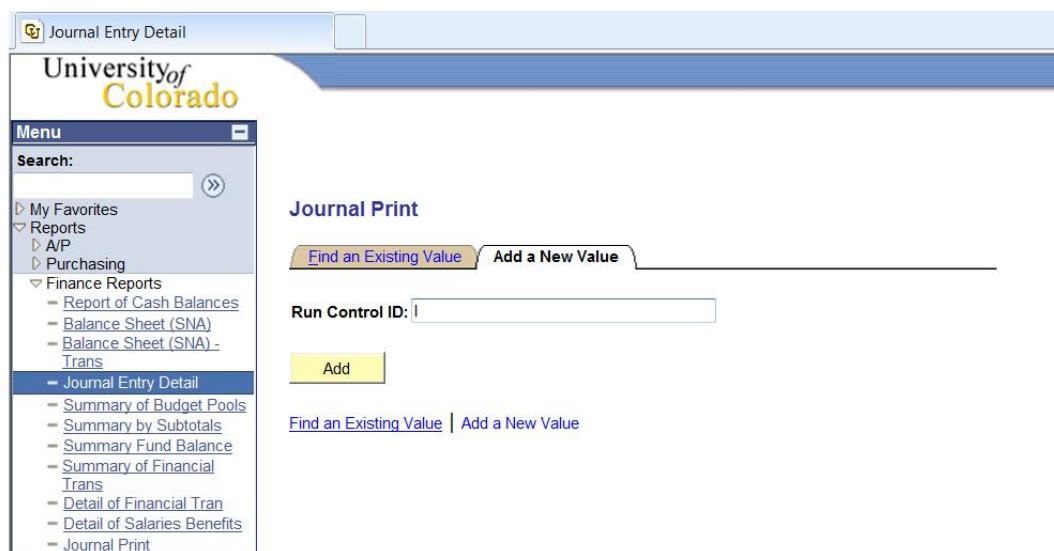
From the “Financial Reports” menu, select “Journal Entry Detail”

1.



2. Enter a run control ID of your choice and click the “Add” button

1. Select the “Add a New Value” tab



1. Enter journal ID

2. Click the “Run” button

3. Enter journal date. The magnifying glass icon can be used to find the date, given the journal ID.

Journal Entry Detail

University of Colorado

Menu

Search:

My Favorites

Reports

A/P

Purchasing

Finance Reports

Report of Cash Balances

Balance Sheet (SNA)

Balance Sheet (SNA) - Trans

Journal Entry Detail

Summary of Budget Pools

Summary by Subtotals

Summary Fund Balance

Summary of Financial Trans

Detail of Financial Tran

Detail of Salaries Benefits

Journal Print

Journal Print

Run Control ID: |

Report Manager

Process Monitor

Run

*Business Unit UCOLO

*Journal ID FMJ139

*Journal Date 11/30/10

Save

Add

Update/Display

1. Then click on the "Reporting Tool" menu to expand it
2. Click the "OK" button

Journal Entry Detail

University of Colorado

Menu

Search:

My Favorites

Reports

A/P

Purchasing

Finance Reports

Report of Cash Balances

Balance Sheet (SNA)

Balance Sheet (SNA) - Trans

Journal Entry Detail

Summary of Budget Pools

Summary by Subtotals

Summary Fund Balance

Summary of Financial Trans

Detail of Financial Tran

Detail of Salaries Benefits

Journal Print

GL A-M

GL N-Z

Sponsored Projects

Set Up Financials/Supply Chain

General Ledger

Accounts Payable

Purchasing

Vendors

Reporting Tools

PeopleTools

Process Scheduler Request

User ID: B02604

Run Control ID: |

Server Name:

Run Date: 04/07/2011

Recurrence:

Run Time: 11:23:30AM

Time Zone:

Reset to Current Date/Time

Process List

| Select | Description | Process Name | Process Type | *Type | *Format | Distribution |
|-------------------------------------|---------------|--------------|--------------|-------|---------|------------------------------|
| <input checked="" type="checkbox"/> | Journal Print | JRNLPURCU | SQR Report | Web | PDF | Distribution |

OK

Cancel

Select "Report Manager"

Favorites

Web Slice Gallery

Base Navigation Page

University of Colorado

Menu

Search:

My Favorites

Reports

Set Up Financials/Supply Chain

General Ledger

Accounts Payable

Purchasing

Vendors

Reporting Tools

Report Manager

Process Monitor

PeopleTools

Reporting Tools

Run, create, and manage queries and nVision reports.

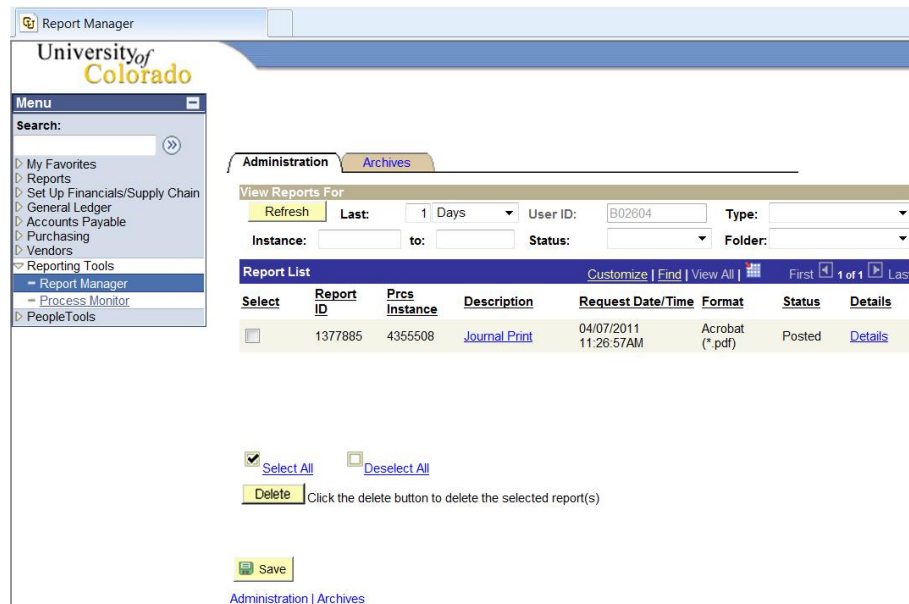
Report Manager

Review report list.

Process Monitor

Review process and

1. Your report will be added to the report list below, but it usually is not available right away. Refresh this page as needed to get an update on the status of your report.
2. When your report becomes available, a hyperlink (in blue underlined font) will appear. Click on it to view your report.



11. Explain Peoplesoft Enterprise HRMS?

PeopleSoft Enterprise HRMS Overview

PeopleSoft HRMS is an integrated suite of applications and business processes that are based on PeopleSoft's Pure Internet Architecture (PIA) and enterprise portal technologies. The sophisticated features and collaborative, self-service functionality available in PeopleSoft HRMS enable you to manage your human resources from recruitment to retirement while aligning your workforce initiatives with strategic business goals and objectives. Types of Peoplesoft Enterprise HRMS is

- PeopleSoft Enterprise HRMS integrations.
- PeopleSoft Enterprise HRMS implementation.

PeopleSoft Enterprise HRMS Integrations

PeopleSoft HRMS integrates with other PeopleSoft applications, such as PeopleSoft Enterprise Financials, PeopleSoft Enterprise Workforce Analytics, and PeopleSoft Enterprise Learning Management. PeopleSoft HRMS also integrates with other third-party applications. PeopleSoft HRMS uses various integration technologies to send and receive data. We discuss integration considerations in the implementation chapters in the PeopleBooks. Supplemental information about third-party application integrations is located on the My Oracle Support website.

PeopleSoft Enterprise HRMS Implementation

PeopleSoft Setup Manager enables you to generate a list of setup tasks for your organization based on the features that you are implementing. The setup tasks include the components that you must set up, listed in the order in which you must enter data into the component tables, and links to the corresponding PeopleBook documentation.

PeopleSoft HRMS also provides component interfaces to help you load data from your existing system into PeopleSoft HRMS tables. Use the Excel to Component Interface utility with the component interfaces to populate tables.

This table lists all of the components documented in the PeopleSoft Enterprise HRMS Application Fundamentals People Book that have component interfaces:

| <i>Component</i> | <i>Component Interface</i> |
|--|-----------------------------------|
| Departments component (DEPARTMENT_TBL) | DEPARTMENT_TBL |
| Establishment component (ESTABLISHMENT_DATA) | ESTABLISHMENT_DATA |
| Job Code Task Table component (JOB_CODE_TASK_TABLE) | JOB_CODE_TASK_TABLE |
| Job Code Table component (JOB_CODE_TBL) | CI_JOB_CODE_TBL |
| Job Tasks component (JOB_TASK_TABLE) | JOB_TASK_TABLE |
| Location component (LOCATION_TABLE) | LOCATION_TABLE |
| FLSA Calendar Table (FLSA_CALENDAR) | FLSA_CALENDAR |
| Pay Run Table (PAY_RUN_TABLE) | PAY_RUN_TABLE |
| Tax Location Table component (TAX_LOCATION_TBL) | TAX_LOCATION_TBL |

Department of Computer Science and Engineering

Subject Name: **ENTERPRISE SOLUTIONS**

Subject Code: **CS T61**

UNIT – V

Siebel Enterprise Applications - Siebel eBusiness Applications – Siebel Tools – Tables and Columns – Business Component – Business Objects – Applets – Joins – Links – Views – Screens – Configuring applications.

2 MARKS

1. What is Siebel?

Siebel is a prominent vendor of interoperable e-business software. They also call themselves Siebel Systems. The company's customer relationship management (CRM), enterprise resource management (ERM), and partner relationship management (PRM) applications are designed to automate those aspects of business and allow an enterprise to perform and coordinate associated tasks over the Internet and through other channels, such as retail or call center networks.

2. Name some customers of Siebel?

Siebel's customers include Chase Manhattan Bank, Deutsche Telecom, IBM, Lucent, Yahoo, and Microsoft.

3. What is Siebel Enterprise Application Integration?

Siebel Enterprise Application Integration (Siebel EAI) is the set of products on the Siebel Business Platform that includes tools, technologies, and prebuilt functional integrations that facilitate application integration.

4. What are the Features of Siebel EAI?

Some of the features provided by Siebel EAI include:

- Provides components to integrate Siebel Business Applications with external applications and technologies within your company
- Works with third-party solutions, such as solutions from IBM, TIBCO, WebMethods, and so forth
- Provides bidirectional, real time and batch solutions to integrate Siebel applications with other applications
- Provides a set of interfaces that interact with each other and with other components within a Siebel application.

5. What is Siebel e-Business applications?

Siebel e-Business applications synchronize the data through one central repository regardless of the contact points (e.g., web, e-mail, call center, field visits and resellers). By having everyone in the organization working with the same repository (data are stored only once), a complete 360-degree view of the customer can be provided to every department or function on an as-needed basis with consistent and up-to-date information available throughout the organization.

6. What is CRM?

Customer Relationship Management (CRM) processes is an approach to managing a company's interaction with current and future customers. It often involves using technology to organize, automate, and synchronize sales, marketing, customer service, and technical support.

7. Write the three levels of CRM?

- Operational CRM (communication strategy level)
- Analytical CRM (marketing strategy level)
- Collaborative CRM (business strategy level)

8. What is Siebel Tool?

Siebel Tools is an integrated environment for configuring all aspects of a Siebel application so a single configuration can be:

- Deployed across HTML and wireless clients
- Used to support multiple Siebel applications and languages
- Easily maintained
- Automatically upgraded to future Siebel product releases

9. Whether Siebel Tools is a programming environment?

Siebel Tools is not a programming environment

10. What are the two windows of Navigation in Siebel Tools ?

- Object Explorer window
- Object List Editor window

11. Define Tables?

A table object definition is the direct representation of a physical database table in a DBMS. Siebel applications provide a set of standard tables that are included in standard Siebel applications. These tables have predefined names and structures.

12. Define Base Table?

The term base table is used in two different contexts:

- The base table for an extension table is the table it extends. This is specified in the Base Table property of the extension table's object definition.
- The base table for a business component is the table that provides most of its essential fields. This is specified in the Table property of the Business Component object definition.

13. Write the Properties of the Table Object Type?

- Name
- Type
- Base Table
- User Name
- Alias

14. Define Data tables?

Data tables comprise most of the tables in Siebel applications. They serve as base tables for business components, and their columns provide the data for fields. Data tables can be public or private.

15. Define an Extension table?

An extension table provides additional columns to a data table that cannot be directly added to the original table because the underlying DBMS may support only a limited number of columns, or will not allow adding a column to a table once it is populated with data. An extension table allows you to provide additional columns for use as fields in a business component without violating DBMS or Siebel application restrictions.

16. Define an Interaction Table?

An intersection table implements a many-to-many relationship between two business components.

17. What is Column object?

A Column object definition is the direct representation of a database column in a DBMS. The name, data type, length, primary key and foreign key status, alias, and other properties of the database column are recorded as properties in the corresponding column object definition.

18. What are the types of Column Objects ?

- Name
- Default
- Physical Type (Physical Type Name)

19. What are the types of Columns?

- Data Columns
- Extension Columns
- System Columns

20. Define Data Columns?

Data columns comprise most of the columns in Siebel applications. They are sometimes referred to as base columns. Data columns provide the data for fields, or serve as foreign keys that point to rows in other tables. The developer cannot modify the properties of data columns, unlike extension columns. Data columns can be public or private.

21. Define Extension Columns?

An extension column is a column that is not used by standard Siebel applications. There are three kinds of extension columns:

- Standard extension columns.
- Custom extension columns in an extension table
- Custom extension columns in a base table

22. Define System Column?

System columns have a value of System in their Type property. System columns appear in all tables in Siebel applications, although the same set of system columns does not appear in every table. You can use the data in system columns for various purposes; for example, the ROW_ID column in tables is used in the construction of joins.

23. What is meant by Business Component?

A business component is a logical entity that associates columns from one or more tables into a single structure.

24. Define Business Object?

A business object implements a business model (as represented in a logical database diagram), tying together a set of interrelated business components using links. The links provide the one-to-many relationships that govern how the business components interrelate in the context of this business object.

25. Define Applets?

An Applet is a data entry form, composed of controls, that occupies some portion of the Siebel application user interface. An applet can be configured to allow data entry, provide a table of data rows, or display business graphics, or a navigation tree. It provides viewing, entry, modification, and navigation capabilities for data in one business component.

26. Write the types of Applets?

- Form Applet
- List Applet

27. Define Form Applet?

A form applet displays data in a data entry form. Fields in the business component appear on the form applet as text boxes, check boxes, and other standard controls

28. Define List Applet?

A list applet allows the simultaneous display of data from multiple records. A list applet displays data in a list table format, much like a spreadsheet or word processor table.

29. Define Joins?

A Join object definition creates a relationship between a business component and a table other than its base table. The join allows the business component to use columns from that table. The

join uses a foreign key in the business component to obtain rows on a one-to-one basis from the joined table, even though the two do not necessarily have a one-to-one relationship.

30. Define Links?

A link implements a one-to-many (or master-detail) relationship between business components based on their base tables. The Link object type makes master-detail views possible, in which one record of the master business component displays with many detail business component records that correspond to the master.

31. What is Multi value Link?

A multi-value link implements a special use of the Link object type, which is the maintenance by the user of a list of records attached to a control or list column in an applet. The group of attached detail records is called a multi-value group.

32. Define Views?

A view is a collection of applets that appear at the same time on the same screen. A view can be thought of as a single window's worth of data forms (applets). Generally, a Siebel application window displays one view at any one time. The currently active view is changed by selecting a different view from the view tabs.

33. Define List Form View?

A list applet and a form applet display data from the same business component. The list applet appears above the form applet. The form applet presents the same information as the currently selected record in the list applet, with a different arrangement that may include more fields.

34. Define Master-detail view?

In a master-detail view, a form applet and a list applet display data from two business components related by a link. The form applet appears above the list applet. The form applet displays one record from the master business component in the master-detail relationship. The list applet displays all of the records from the detail business component that have as their master record the record currently displayed in the form applet.

35. Define Thread Bar?

The thread bar is a navigational tool for the user. It provides the means to navigate from view to view among the views previously visited in the current screen.

36. What is Drilldown Behavior in a View?

It allows the user to drill down from a cell in a list applet (or using a pop-up menu in either a form or list applet) to a particular view. Drilldown controls or list columns in a list applet in Siebel applications consist of colored, underlined text, much like a hypertext link in a Web browser.

37. Give the types of Drilldown Behavior?

- Static Drilldown Behavior
- Dynamic Drilldown Behavior

38. Define screens?

A screen is a logical collection of views. It is not a visual construct in itself; rather, it associates views so that other visual constructs, such as the Site Map and tab bar, can reflect the list of views contained in the currently active screen.

39. Write any Configuring applications for deployment in High Interactivity mode.

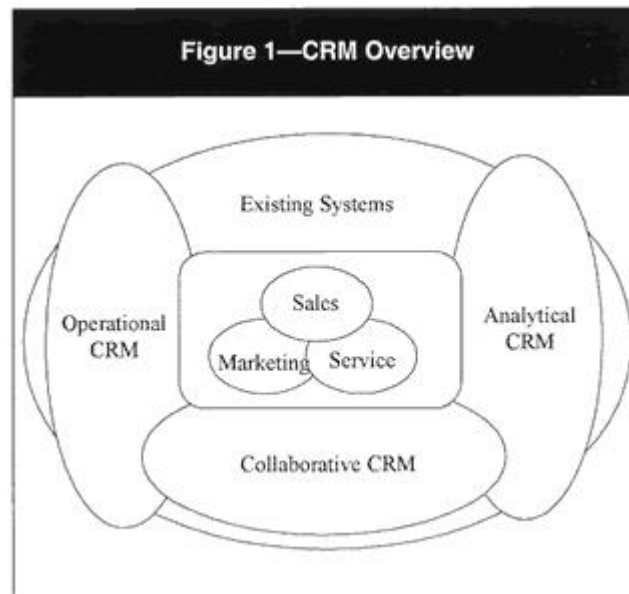
- Browser scripting is fully supported in High Interactivity mode.
- You cannot modify the appearance of the rich text editor.
- You cannot modify the background and text color of list applets.
- You cannot place method-invoking controls.

11 MARKS**1. Explain CRM Processes in detail.**

Companies, both large and small, are looking to gain resources like automating finance, manufacturing, distribution, human resources management and general office operations, many businesses through CRM strategies that enable them to manage all customer contact points, including person-to-person contact, the telephone, e-mail and the Internet.

Different levels of CRM³ (see **figure 1**) include:

- **Operational CRM** (communication strategy level) is the automation of the customer-oriented front-end marketing, sales and service processes and their integration with back-office systems. An example of operational CRM is a customer database maintained purely for direct mailing purposes.
- **Analytical CRM** (marketing strategy level) is the analyses of the data created with operational CRM to better meet the needs and desires of customers. For example, relations between events and buying behaviors are analyzed, typically in a data warehouse (possibly with neural network capabilities).
- **Collaborative CRM** (business strategy level) is the overall communication and coordination model of a customer life cycle between the channels and contact points. In this collaborative CRM approach, business processes and a central customer database are drivers for success.



Siebel is designed to deal with all the different levels of CRM and supports CRM as a business strategy. It should be taken into account that the way CRM is implemented and effectively utilized will vary between organizations. It is dependent on many variants, including the way of doing business, buying habits of customers (e.g., convenience vs. specialty goods), industry (e.g., telecom with many consumers vs. forestry with a few business customers), company size, existing systems and organization structure.

Organization-specific aspects need to be incorporated in the process designs for marketing (e.g., campaign management, lead generation), sales (e.g., opportunity management, contact and account management and quotation management) and service (e.g., call center processes, service agreement management). These processes have to be related to the relevant business units and Siebel products. As a result of this mapping process, changes or improvements may be identified for the business units as well as gaps between the process design and the Siebel standard solution.

It is important for the IT auditor to be involved in the process design to ensure integrity, security, auditability, controllability, availability, efficiency and effectiveness requirements are defined for the core marketing, sales and service processes as well as for the management control and support activities surrounding these processes (e.g., interfaces with the backbone ERP software).

2. Explain Siebel Tool in detail?

Siebel Tools is an integrated environment for configuring all aspects of a Siebel application so a single configuration can be:

- Deployed across HTML and wireless clients
- Used to support multiple Siebel applications and languages

- Easily maintained
- Automatically upgraded to future Siebel product releases

Siebel Tools is not a programming environment; it is a declarative application configuration tool. Standard Siebel applications provide a core set of object definitions that you can use as a basis for your tailored application. Using Siebel Tools and other configuration tools that are part of a Siebel solution, Siebel application developers, system administrators, and database administrators can customize a standard Siebel application without modifying source code or SQL. Some of the configuration tools are accessed through the Siebel applications. Siebel Tools, however, is a separate product with its own user interface.

Navigation in Siebel Tools is done mainly in two windows:

- Object Explorer window
- Object List Editor window

3. Explain in details about Tables.

Tables:

A table object definition is the direct representation of a physical database table in a DBMS. Siebel applications provide a set of standard tables that are included in standard Siebel applications. These tables have predefined names and structures, and typically begin with one of the following prefixes:

S_ Table names starting in S_ are standard tables for supplied with Siebel Sales and Siebel Service. For example, the S_CONTACT table stores contact information, and S_OPTY stores opportunity information. Nearly all standard tables are of this type.

W_ Table names starting in W_ are Data Warehouse tables used in Siebel Analytics to demoralize data used in the S_ tables.

Base Tables

The term base table is used in two different contexts:

- The base table for an extension table is the table it extends. This is specified in the Base Table property of the extension table's object definition.
- The base table for a business component is the table that provides most of its essential fields. This is specified in the Table property of the Business Component object definition. The set of fields supplied by the base table is supplemented by non updateable fields that are obtained from joins.

Properties of the Table Object Type:

The following are the key properties in a table object definition:

- 1. Name:** Provides the name of the table in the DBMS.
- 2. Type:** Indicates which of the following styles describes the table.

- **Data (Public).** Public data tables are among the original set of tables implemented in Siebel applications. They hold data that is made available through business components to developers and users. Public data tables can be extended using extension tables and extension columns. These extension are subject to database restrictions. Data tables are discussed in “Data Tables” on page 196.
- **Data (Private).** Private data tables are similar to public data tables, but cannot have extension columns.
- **Data (Intersection).** Identifies an intersection table. An intersection table implements a many-to-many relationship between two data tables. Intersection tables are discussed in “Intersection Tables” on page 204.
- **Extension.** An extension table adds additional columns to a data table that the original data table is unable to hold due to DBMS platform or Siebel application design restrictions. Extension table names have an _X suffix, or _XM or one-to-many, or _T for TAS extension tables. NOTE: Extension tables that have _XM suffix have table type of “Data (Public).” Extension tables are discussed in “Extension Tables” on page 196.
- **Interface.** Interface tables are used by Siebel Enterprise Integration Manager (EIM) to import initial data for populating one or more base tables and subsequently to perform periodic batch updates between Siebel applications and other enterprise applications. Interface table names end in _IF or _XMIF. Interface tables are discussed in “Column Objects” on page 215.
- **Database View, Dictionary, Journal, Log, Repository, Virtual Table, and Warehouse styles.** These are all table types that are reserved for Siebel internal use.
- **Extension (Siebel).** These tables are reserved for Siebel use only. They are usually extensions from S_PARTY. If customers want to extend person- and organization-related tables they need to extend from S_PARTY. For example, S_CONTACT is an extension table of S_PARTY. Because S_CONTACT is of type Extension (Siebel), you cannot use it as a parent table for an extension table. You must use S_PARTY. For a business component based on your new table to show data from S_CONTACT, you must create a Join object that references S_CONTACT and has a Join Specification child object with a Source Field property set to Parent Id and Destination Column property set to ROW_ID. The row ID of an S_CONTACT record will be the same as the row ID of the corresponding S_PARTY record.

3. Base Table. Identifies the base table if the table in the object definition is an extension table. If the table in the object definition is a base table, this property is blank. An extension table always identifies a base table.

4. User Name. A longer, descriptive name that aids in identifying the table when used in configuration.

5. Alias. A name that can be used as a synonym for the table name to make the name more understandable. For example, an alias such as S_Organization_External could be specified for the S_ORG_EXT table.

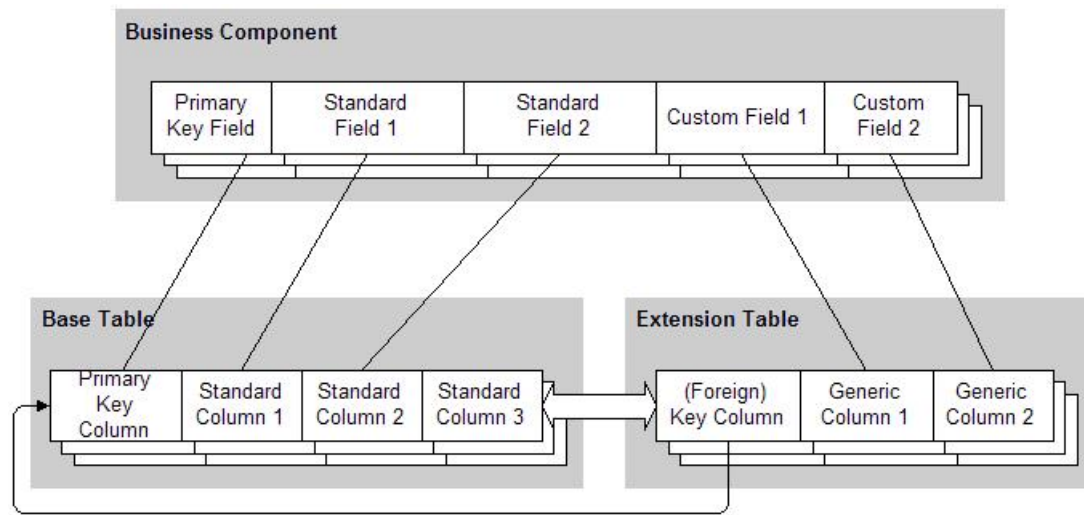
Data Tables :

Data tables comprise most of the tables in Siebel applications. They serve as base tables for business components, and their columns provide the data for fields. Data tables can be public or private.

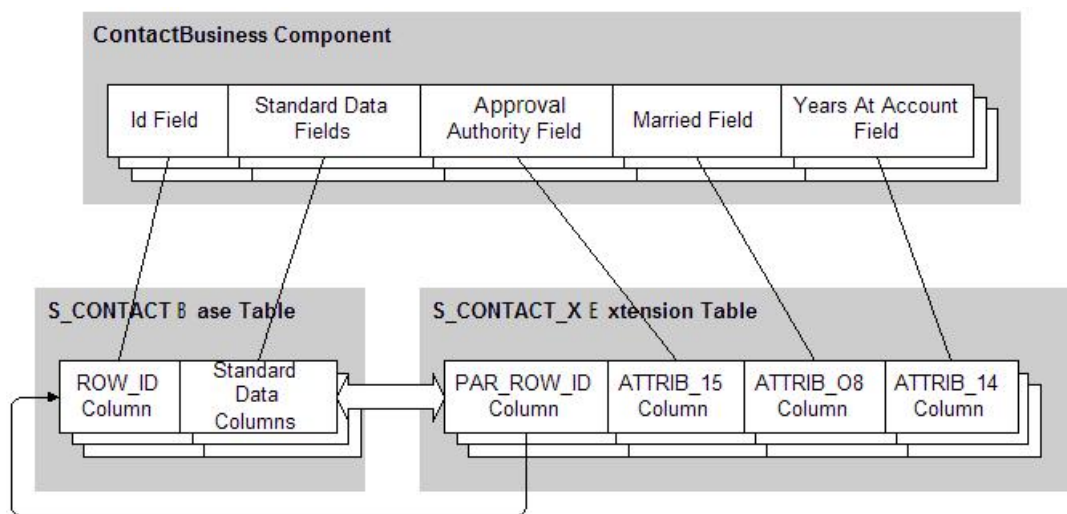
4. Explain in detail about Extension Tables.

An extension table provides additional columns to a data table that cannot be directly added to the original table because the underlying DBMS may support only a limited number of columns, or will not allow adding a column to a table once it is populated with data. An extension table allows you to provide additional columns for use as fields in a business component without violating DBMS or Siebel application restrictions.

An extension table is a logical augmentation of an existing table. Its columns are provided mostly for developers, and are generally not used by standard Siebel applications. An extension table extends a base table in the sense that it effectively adds additional columns. These columns are not physically part of the base table, but are available for use in a business component alongside the base table columns as if they were. When columns in a base table are updated, the time stamps of its extension tables are not updated unless columns in those extension tables are also updated. When records in an extension table are changed, system columns in a parent table are updated. This is done because the associated record in an extension table is considered by the object manager to be logically a part of its parent record. The relationships between a base table, an extension table, and the business component that uses them are given below in the figure.



Business Component, Base Table and Extension Table



Extension Table Example

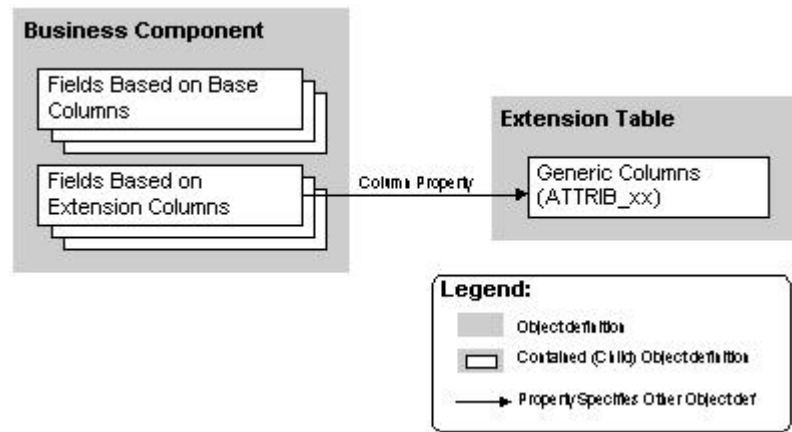
Extension tables can be of the one-to-one or one-to-many style:

Rows in one-to-one extension table have a one-to-one relationship with corresponding rows in the base table. A one-to-one extension table extends the base table horizontally

In a **one-to-many** extension table, there are multiple extension table rows for each base table row. There are standard one-to-many extension tables for certain of the major business components, including Opportunity, Contact and Account. These are used primarily to create multi-value groups based on user-created business components.

One-to-One Extension Tables

One-to-one extension tables have the _X suffix on their names (with the exception of TAS tables, which have the suffix _T). The details of the object definition relationships (excluding the implied join) are given in the figure.



Extension Table Details (Excluding Implied Join)

The object definitions in the above figure are:

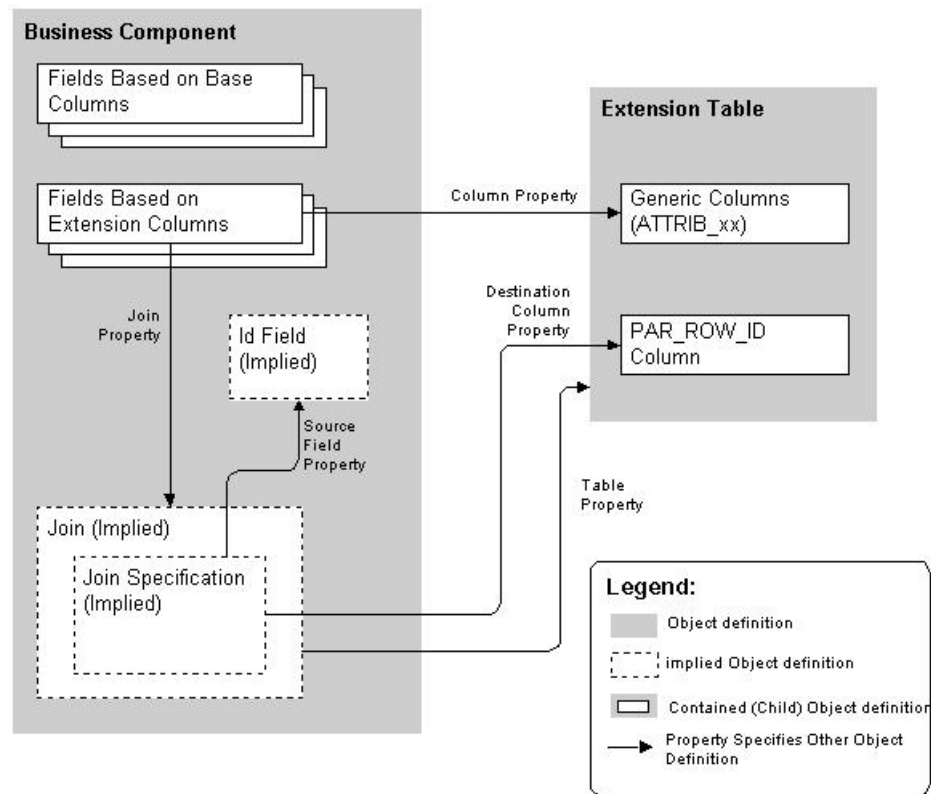
- **Business component.** Business component being extended.
- **Fields based on base columns.** Fields that represent data from columns in the business component's base table. They are unaffected by the extension table.
- **Fields based on extension columns.** Represent data from columns in the extension table.
- **Extension table.** Provides columns that may be used to add developer-defined fields to the business component.

Implied Joins

Underlying the one-to-one extension table's relationships with the base table and business component is a set of hidden relationships called an *implied* or *implicit join*. The implied join makes the extension table rows available on a one-to-one basis to the business component that uses the extension table. Every extension table has an implied join with the business component it extends. This join always has the name of the extension table.

An implied join is different from joins defined as object definitions. Data can be updated through an implied join. Data can be displayed only through other joins. This update capability is important for extension table functionality.

When a field in the business component is based on a column in the extension table, the Column property of the Field object is set to the name of the column, and the Join property is set to the name of the extension table. For example, the Birthday field in the Contact business component has a Column property value of ATTRIB_26 and a Join property value of S_CONTACT_X.



Extension Table Details with Implied Join

5. Explain Intersection Table with neat architecture.

An *intersection table* implements a many-to-many relationship between two business components.

A *many-to-many relationship* is one in which there is a one-to-many relationship from either direction. For example, there is a many-to-many relationship between Opportunities and Contacts. One Opportunity can be associated with many Contact people, and one Contact person can be associated with many Opportunities.

The below figure shows the Account Detail - Contacts View, in which one account is displayed with multiple detail contacts.

Account

New Save 2 of 7+

*Name: 3Com Main Phone #: (773) 326-5000 Partner: ☐ Industries: manufacturing industries

Address Line 1: 7074 N Clark St City: Chicago Competitor: ☐ Territories:

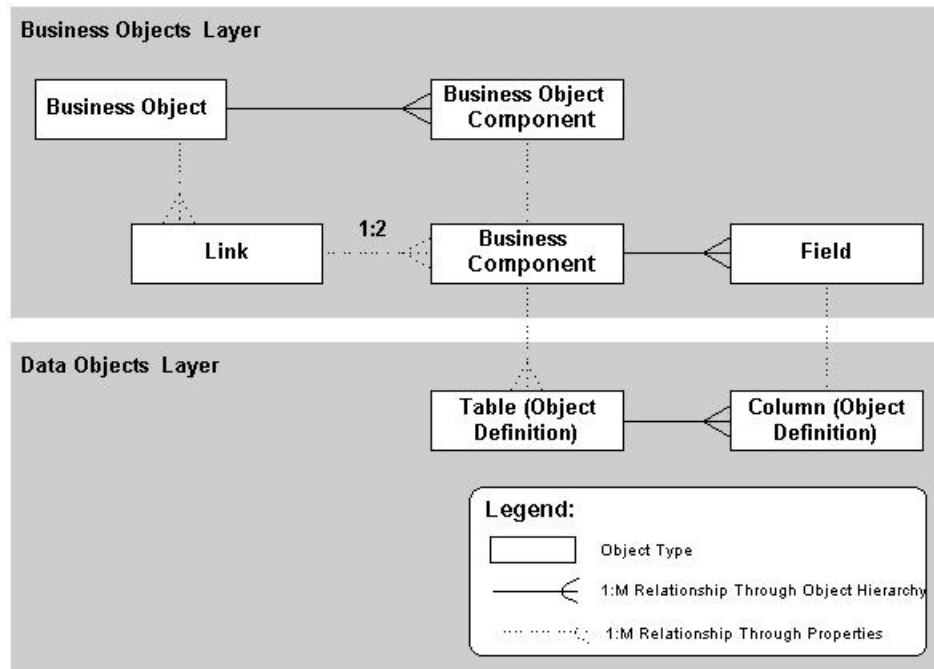
Address Line 2: State: IL Account Type: Customer Status: Gold Account Team: SADMIN

Zip: 60626 Country: USA Parent:

Calendar Categories Contacts Contact Admin Entitlements Messages Notes Opportunities Orders Organization

New Save 1 - 7 of 7+

| New | Last Name | First Name | Mr Ms | Work Phone # | Job Title | Email |
|-----|-----------|------------|-------|----------------|----------------------|------------------------|
| | Ahl | Avram | Mr. | (773) 555-9870 | Vice President | a.ahl@3Com.com |
| | Genever | Alan | Mr. | (773) 555-6500 | Manager, Account F | alan.genever@3Com.com |
| | Helena | Kristy | Ms. | (773) 555-2223 | Supervisor, Logistic | kristy.helena@3Com.com |
| ✓ | Ishii | Jill | Ms. | (773) 555-5312 | Account Manager | jill.ishii@3Com.com |
| | Kreisberg | Bob | Mr. | (650) 786-8767 | Programmer | bob.kreisberg@3Com.com |
| | Mann | Elisa | Ms. | (773) 355-8675 | Service Manager | elisa.mann@3Com.com |
| | Navarro | Alan | Mr. | (773) 555-6541 | Programmer | alan.navarro@3Com.com |



Intersection Table Architecture

The following are descriptions of the object definitions:

Business object. The business object references the link (indirectly through the business object's child business object component) that uses the intersection table. It also contains the two business components included in the link.

Business object components. Business Object Component object definitions are used to include business components in the business object. Business Object Component is a child object type of Business Object. The detail business object component references both the detail business component, by means of the Business Component property, and the link, by means of the Link property. The master business object component only references its corresponding business component.

Link. The link object definition establishes a one-to-many relationship between the two business components in a particular direction. That is, the property settings in the link specify that one business component is the master and the other is the detail in the master-detail relationship.

Master and detail business components. The two business components are specified in the link. They provide data to the user interface object definitions that display the master-detail relationship. The base table of each business component contains the ROW_ID column referenced by the Inter Child Column (detail) and Inter Parent Column (master) properties of the Link object type.

Intersection table. The intersection table holds the associations between rows in the base tables of the master and detail business components. Each row in the intersection table represents one association between the two business components. Two columns in the intersection table serve as foreign keys to the base tables of the two business components.

These columns are identified in the Inter Parent Column and Inter Child Column properties of the link.

Inter Parent column. This column in the intersection table holds the pointer to the associated row in the master business component's base table. It is identified in the Inter Parent Column property of the Link object.

Inter Child column. This column in the intersection table holds the pointer to the associated row in the detail business component's base table. It is identified in the Inter Child Column property of the Link object.

ROW_ID columns. The base table of each business component has a unique identifier column for the rows in that table. This is the ROW_ID column

Column Objects

A Column object definition is the direct representation of a database column in a DBMS. The name, data type, length, primary key and foreign key status, alias, and other properties of the database column are recorded as properties in the corresponding column object definition.

A column has one of several styles based on the value in the Type property. These styles include Data, Extension, IFMGR, System, and others.

Column Object Type

The column object corresponds to one database column in the database table that is represented by the parent table object definition. Each database column in the database table needs to have a corresponding column object definition. The important properties of the Column object type are as follows:

Name. Provides the name of the database column in the database table.

Default. Provides a default value when new rows of this table are added.

Physical Type (Physical Type Name). Identifies the data type of the column in the database. The following data types are supported:

Character. Used for fixed-length text. Also used for Boolean columns, which are character columns with a length of 1.

Long. Long text. You can store approximately 16K worth of data in long columns. By default, you cannot have char greater than 1. If you want, you need to set the preference under Options/Database.

Varchar. Variable-length text. Used for memo-type fields and to store row-ID and foreign key values.

Number. Any numeric data. Typical numeric columns in Siebel applications are 22,7 for general-purpose numbers, and 10,0 for integers.

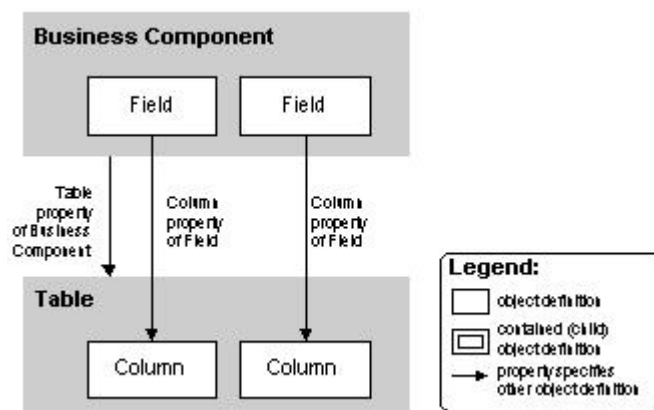
6. Describe Business Components in detail.

A *business component* is a logical entity that associates columns from one or more tables into a single structure. Business components provide a layer of wrapping over tables, so that applets reference business components rather than the underlying tables. This creates convenience (all associated columns are together in one bundle), developer-friendly naming, and the separation of the developer role from the database administrator role. A business component can also have a default sort or search specification, providing records to applets in a predetermined sort order and according to a selection criterion. When instantiated in a Siebel application, a business component is comparable to a recordset. Its definition in Siebel Tools provides the foundation for controlling how data is selected from, inserted, and updated within the tables it references.

| New | Name | Site | Main Phone # | Territories | Industries | Status | URL |
|-----|----------------------|-------------|----------------|-------------|----------------------|--------|-------------------|
| | Woolen Goodrich N | Boston | (617) 244-5020 | | animal specialties | Active | |
| | 3Com | Santa Clara | (408) 326-5000 | | manufacturing indus | Gold | www.3com.com |
| | AMCO Communicati | Chicago, IL | (847) 491-2300 | | steel pipe & tubes | Active | www.amco.net |
| * | Alliance Program | | | | | | |
| | Andercotti Consultin | Palo Alto | (415) 389-5400 | | services | Active | |
| | Andronico | Berkeley | (510) 568-1200 | | groceries & related | Active | |
| | BO Edwards, Inc | HQ | (212) 489-1500 | | theatrical producers | Active | www.agedwards.com |

Business Component Records in a List Applet

A business component contains fields. Each column whose data is included in a business component is represented with a field. The Field object type is a child of the Business Component object type. The Column object type is a child of the Table object type. The set of field child object definitions of a business component maps a corresponding set of columns into the business component.



Business Component Field and Column Relationships

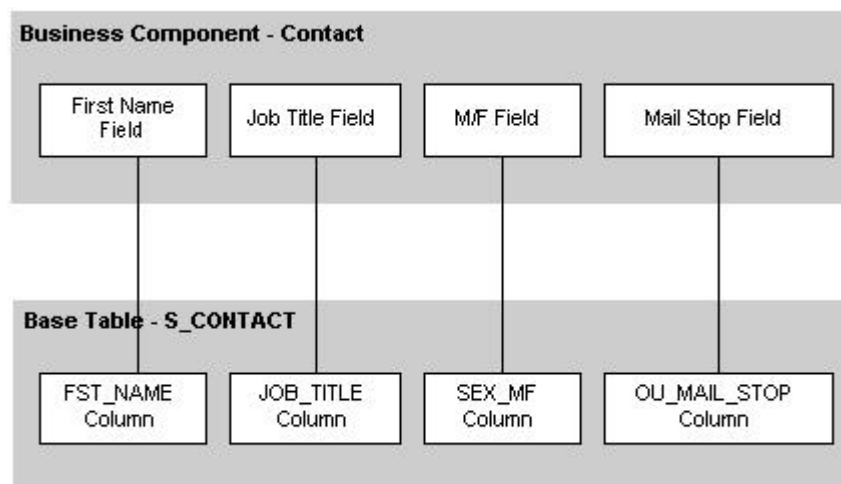
In addition to connecting a column to the business component, the Field object type also allows you to specify a meaningful name that refers to that column. Columns in tables are often cryptically named to match the names in the DBMS, whereas fields can have more

meaningful and longer names than the columns they represent (75 characters long as opposed to, typically, 30 characters).

Base Tables of Business Components

A *base table* of a business component is assigned to the business component to provide the most important columns for use as fields in the business component. Fields built on the base table can be edited, whereas fields built on joined tables can only be displayed.

The base table is assigned to the business component with the Table property in the Business Component's object definition. The below is an example of some fields in the Contact business component that map corresponding columns from the business component's base table, S_CONTACT.



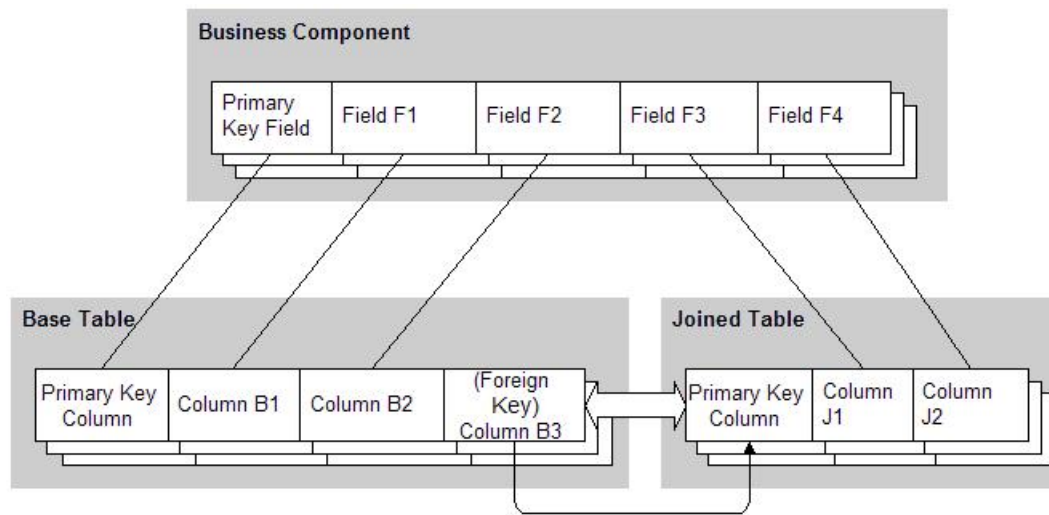
Examples of Fields Representing Columns

Every business component has a base table assigned to it. It is not essential that the business component include all of the columns in the base table, although typically it will include most of them. In particular, system columns in the base table such as ROW_ID, CREATED_BY and LAST_UPD_BY are automatically represented in the business component through implied fields. System columns do not require field object definitions in the business component.

Joined Tables

A *joined table* provides rows on a one-to-one basis to the business component as a result of a foreign key relationship between the joined table and the business component's base table. That is, for every record in the business component (which corresponds to a row in the base table) there can be a corresponding row in the joined table. However, not every record in the base table will have a record in the joined table.

The data obtained by a business component through a join (other than to an extension table) is read-only in that business component.



Fields from the Base Table and a Joined Table

Extension Tables

Extension tables are a special kind of joined table. Like other joined tables, extension tables provide rows on a one-to-one basis in parallel with base table rows. Extension tables are identified by the `_X` suffix in the table name, such as `S_ORG_EXT_X`, which extends `S_ORG_EXT`.

Extension tables are provided specifically to allow columns to be virtually added to a base table rather than physically added. This provides the means to expand base tables without violating DBMS or database design restrictions, and without the need to perform complicated database restructuring operations. Extension table data, unlike the data in other joined tables, can be updated in the business component.

Intersection Business Components

An *intersection business component* is a business component based on an intersection table. It provides the means to display all of the combinations of data in a many-to-many relationship, instead of only one or the other one-to-many relationship of which it is composed.

Intersection tables implement many-to-many relationships. Some (such as `S_OPTY_CON` and `S_ACCNT_POSTN`) also provide intersection data through a join to one or the other master business component that uses the intersection table. Intersection data is data that resides in columns other than the two required foreign key columns in the intersection table, and is specific to the intersection of the two master business components.

7. Describe in detail about Virtual Business Components:

Business components based on external data are called *virtual business components*. Virtual business components are used when the business component has to obtain data from a location other than a database table in the Siebel database, but the information has to be presented in the standard Siebel user interface (applets and views). This is typically real-time information from another database, such as from the Report Encyclopedia in Actuate, or from an SAP table, although anything that can supply data in response to a SQL query is a candidate.

Virtual business components allow you to:

- Represent external data (for example, data in an SAP R/3 database) as a virtual business component within a Siebel application—the business component configuration specifies the DLL to use to access the data
- Use business services to transfer data

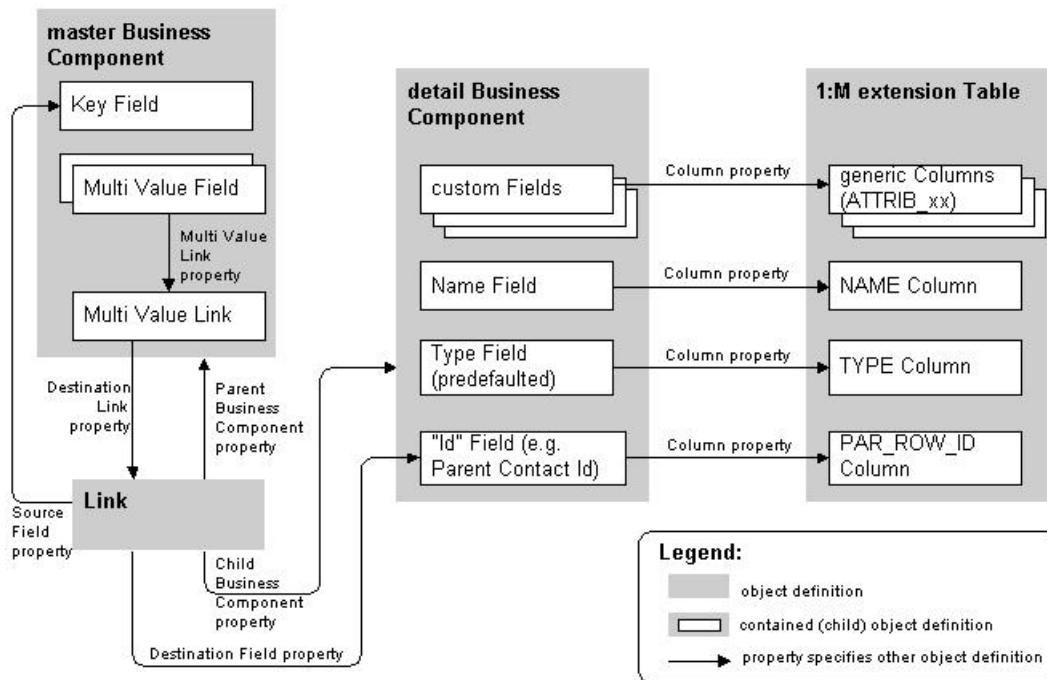
Virtual business components support properties such as:

- Single-value fields
- Field-level validation
- Standard business component event model (for example, PreNewRecord, PreDelete, and so on)
- Insert, delete, query, and update operations

Additional information about virtual business components:

- Applets can be based on virtual business components.
- Virtual business components can be accessed through object interfaces.
- All business component events are available for scripting.
- Virtual business components cannot be docked.
- Virtual business components can be used as stand-alone or children business components in a business object.
- Virtual business components support dynamic applet toggles. For more information about applet toggles.

8. Explain Master-Detail Business Components in detail.



One-to-Many Extension Table Details

Master Business Components

The master business components will hold the new multi-value fields. Master business component contain the following important object definitions:

■ **Key field.** This is the key field in the master business component; it is used to reference individual records. Typically, it is named Id. The Source Field property of the Link object definition points to this field. The property value may be blank because, by default, a blank Source Field value refers to the Id field.

Multi-value field. The multi-value field provides access to a corresponding field value in the current record of the detail business component. The control or list column that displays the multi-value field normally will be able to invoke a multi-value group applet for display and maintenance of the detail records.

■ **Multi-value link.** The multi-value link provides access to the set of records in the detail business component. One multi-value link is created for each multi-value group that is created using the one-to-many extension table.

Link

The link object definition creates the one-to-many relationship between the master and detail business components. There are no special link configuration issues related to one-to-many extension tables.

Detail Business Component

The detail business component represents the one-to-many extension table used by the multi-value link and multi-value group applet. Multiple custom business components can be created using the same one-to-many extension table. Each custom business component presents a different type of data for use in a different multi-value group.

The detail business component contains custom fields that represent generic (ATTRIBxx) extension columns, and hold whatever data is required for the application. For example, an Area Of Expertise business component might have a Subject Area field, a Years Of Experience field, and a Licensed field. Each field is a mapping of a different generic extension column.

The following three fields are part of the User Key (U1), which uniquely identifies a row for EIM:

Name field. The name field represents the NAME column from the extension table. It provides the means for the user to enter an identifying value in each record. For example, in a Hobbies business component, the name field might be called Hobby. The user would enter the name of a hobby into each record in this field.

Type field. The type field is usually named Type, and represents the TYPE column. It contains the same value for all records in one multi-value group, and distinguishes the records of that multi-value group from others. It should be set in the Predefault property to some identifying word or phrase, such as HOBBY, EXPERTISE or PRIOR JOB, and should not be exposed in the user interface.

Parent ID field. The parent ID field represents the PAR_ROW_ID column. Generally it is named Parent Contact Id, Parent Account Id, or something similar. It identifies the row ID of the base table row corresponding to the parent record in the master business component. The parent ID field is specified in the Destination Field property of the Link object.

The detail business component contains one important property for use with a one-to-many extension table:

■ **Search Specification.** The Search Specification property should be set to restrict the records retrieved to only those with a specific value in the Type field. This is the same value that is specified in the Pre Default Value property for that field.

9. Explain Business Objects in Detail.

Business Objects A business object represents a major functional area of the enterprise—every major entity has a business object. Examples of business objects are Opportunity, Account, and Contact. A business object is a collection of related business components; for example (as shown in Figure 6), the Opportunity business object consists of Opportunities plus related Contacts, Activities, Products, and Issues. Each business object has one business component that serves as the master or driving business component. This master business component provides focus for the business object, and they both have the same name (the name is Opportunity).

Business Object Types:

Business Component. A business component is a logical entity that associates columns from one or more tables into a single structure. Business components provide a layer of wrapping over tables, causing applets to reference business components rather than the underlying tables. This design creates convenience (all associated columns together in one bundle), developer-friendly naming, and the isolation of the developer role from the database administrator role. Multiple users can instantiate copies of the same business component. Data changes made by any one user are reflected in all instances of the business component.

Field. A field object definition associates a column to a business component. This is how columns from tables are assigned to a business component and provided with meaningful names that the customer developer can easily change. Alternately, a field's values can be calculated from the values in other fields in the business component. Fields supply data to controls and list columns in the Web interface.

Business Object. A business object implements a business model (logical database diagram), tying together a set of interrelated business components using links. The links provide the one-to-many relationships that govern how the business components interrelate in the context of this business object.

Business Object Component. A business object component object definition is used to include a business component and, generally, a link in the business object. The link specifies how the business component is related to another business component in the context of the same business object. **Link.** A link implements a one-to-many relationship between business components. The Link object type makes master-detail views possible. A masterdetail view displays one record of the master business component with many detail business component records corresponding to the master. A pair of links also may be used to implement a many-to-many relationship.

Multi-Value Link. A multi-value link is used in the implementation of a multi-value group. A multi-value group is a user-maintainable list of detail records associated with a master record. The user invokes the list of detail records from the master record when it is displayed in a list or form applet. For example, in an applet displaying the Account business component, the user can

click the Select button to the right of the Address text box to see a pop-up window displaying multiple Address records associated with the currently displayed account.

Join. A join object definition creates a relationship between a business component and a table that is not the business component's base table. The join allows the business component to build fields using columns from the non-base (joined) table. The join uses a foreign key in the business component to obtain rows on a one-to-one basis from the joined table, even though the two do not necessarily have a one-to-one relationship.

Join Specification. A join specification is a child object type of join that provides details about how the join is implemented within the business component.

User Property. A user property is a temporary storage field in a business component that is not tied to the database. When a user sets a value in a business component user property, it is not visible to other users of the same business component. User properties are not saved across sessions.

Business Service. A business service is a reusable module containing a set of methods. It provides the ability to call its C++ or script methods from customer-defined scripts and object interface logic, through the invoke-method mechanism.

10. Explain Applet and types of applet?

An *applet* is a data entry form, composed of controls, that occupies some portion of the Siebel application user interface. An applet can be configured to allow data entry, provide a table of data rows, or display business graphics, or a navigation tree. It provides viewing, entry, modification, and navigation capabilities for data in one business component.

An applet is always associated with a business component. Although the same business component can be associated with multiple applets, an applet is associated with only one business component.

Applets are associated with one or more Siebel Web templates. Web templates are files that contain HTML and proprietary Siebel tags that define the layout and format of the applet in the user interface.

Types of Applets:

Form applet. A *form applet* displays data in a data entry form. Fields in the business component appear on the form applet as text boxes, check boxes, and other standard controls. A form applet appears in below figure.

The screenshot displays a Siebel form applet titled "Incentive Compensation - 2500 See". The form is organized into several sections:

- Header:** Includes navigation tabs (Calendar, Categories, Competitors, Decision Issues, Estimate Compensation, Messages, Organizations, Partners, Presentations, Products) and a "New Query" button.
- Form Fields:**
 - Name:** Incentive Compensation - 2500 See
 - Description:** (Empty text box)
 - Sales Team:** SADMIN
 - Opportunity Currency:** USD
 - Revenue:** \$2,625,000.00
 - Expected Value:** \$525,000.00
 - Best Case:** \$0.00
 - Worst Case:** \$0.00
 - Probability %:** 20%
 - Sales Stage:** 06 - Short List
 - Status:** (Empty dropdown)
 - Reason:** (Empty dropdown)
 - Committed:** ☒
 - Executive Priority:** ☐
 - Close Date:** 11/30/2002
 - Created:** 3/1/2001 9:34:06 AM

Form Applet

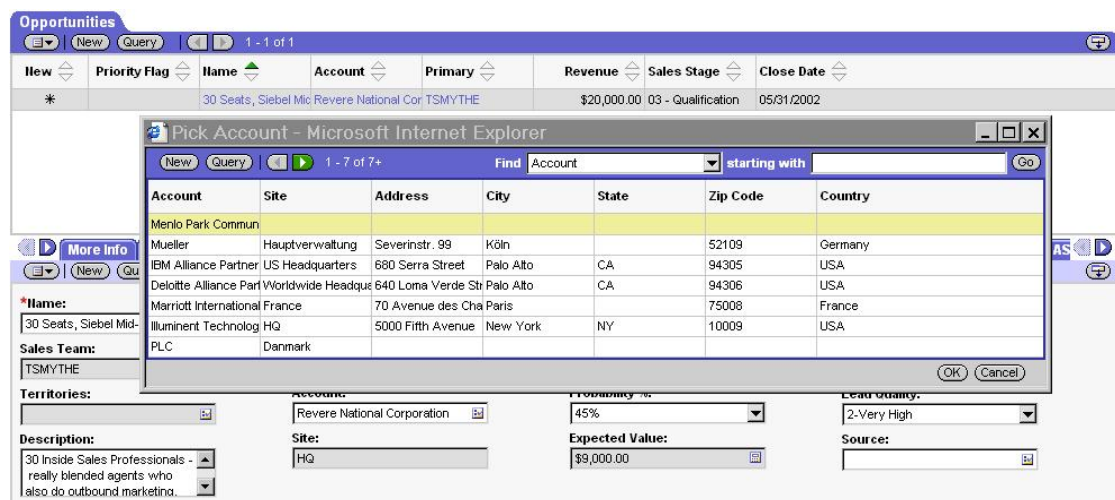
List applet. A *list applet* allows the simultaneous display of data from multiple records. A list applet displays data in a list table format, much like a spreadsheet or word processor table. Rows in the list applet correspond to records in the business component; list columns in the list applet correspond to fields in the business component. In addition to textual data, lists also support images in JPEG and GIF formats and edit controls such as check boxes, drop-down lists, MVGs, and text fields. A list applet appears in below figure.



| New | Name | Site | Main Phone # | Territories | Industries | Status | URL |
|-----|---------------------------|--------------|----------------|-------------|---------------------------------|--------|----------------|
| | 3COM | Santa Clara | (408) 326-5000 | | manufacturing industries | Gold | www.parker.com |
| | 3Com Corporation | Headquarters | (415) 329-6500 | | manufacturing industries | Active | |
| | 3Com Services | US | (415) 329-6500 | | manufactured hardware (general) | Active | |
| | AMCO Communications | San Rafael | (415) 491-2300 | | steel pipe & tubes | Active | |
| | Acme Inc. | HQ | (510) 245-3000 | | advertising agencies | Active | |
| | Acme New England Division | Boston | (617) 232-1121 | | animal specialties | Active | |
| * | Alliance Program | | | | | | |

List Applet

Pick applet. A *pick applet* is a dialog box window that appears when a selection is to be made in a control or list column that has the check mark icon to its right. The pick applet provides a list or table of selection values, from which the user selects a value or record. A pick applet displays data that has a M:1 relationship to the data in the parent applet. A pick applet appears in below figure.



| Account | Site | Address | City | State | Zip Code | Country |
|------------------------|--------------------|--------------------|-----------|-------|----------|---------|
| Menlo Park Commun | | | | | | |
| Mueller | Hauptverwaltung | Severinstr. 99 | Köln | | 52109 | Germany |
| IBM Alliance Partner | US Headquarters | 680 Serra Street | Palo Alto | CA | 94305 | USA |
| Deloitte Alliance Part | Worldwide Headquar | 640 Loma Verde Str | Palo Alto | CA | 94306 | USA |
| Marriott International | France | 70 Avenue des Cha | Paris | | 75008 | France |
| Illuminant Technolog | HQ | 5000 Fifth Avenue | New York | NY | 10009 | USA |
| PLC | Danmark | | | | | |

Pick Applet

Multi-value group applet. A *multi-value group applet* is used for entry, maintenance, and viewing of a list of detail records associated with one or more fields in the currently

displayed master record. MVGs allow the user to associate multiple records to a single field in a form or list and provide a way of representing one-to-many relationships within a single record of data.

There are two ways in which users can add data to the MVG:

The screenshot shows a Siebel CRM interface. At the top, there's a header bar with 'Accounts' and navigation buttons like 'New', 'Query', and '1 - 7 of 7+'. Below this is a list of team members with columns: Primary, Last Name, First Name, Position, Email, Phone #, Manual, and System. The list includes entries for Reilly, Major, Morgan, Administrator, Silver, and Murphy. Below the list is a form for adding data to the MVG. The form has sections for 'Main Phone #', 'Main Fax #', 'Account Team', 'Parent', 'Stage', 'Expertise', 'Reference', and 'PO Approved'. The 'Account Team' section shows 'SADMIN' selected. The 'Stage' dropdown is set to 'Maintenance'. The 'Reference' checkbox is checked. The 'PO Approved' checkbox is unchecked.

Multi-Value Group Applet

Chart applet. A *chart applet* graphically displays data from a business component in a bar chart, line graph, pie chart, scatter diagram or other format. It summarizes and illustrates data relationships. Charts are usually accessed through a tab in the third-level navigation level and contain a number of sub- category views (multiple charts of data). These are displayed in an overview of miniature chart graphics (.gif images) with title text. Both the mini-graphics and the title text for the chart are hyperlinked to the detailed version of the chart. A chart applet appears below.

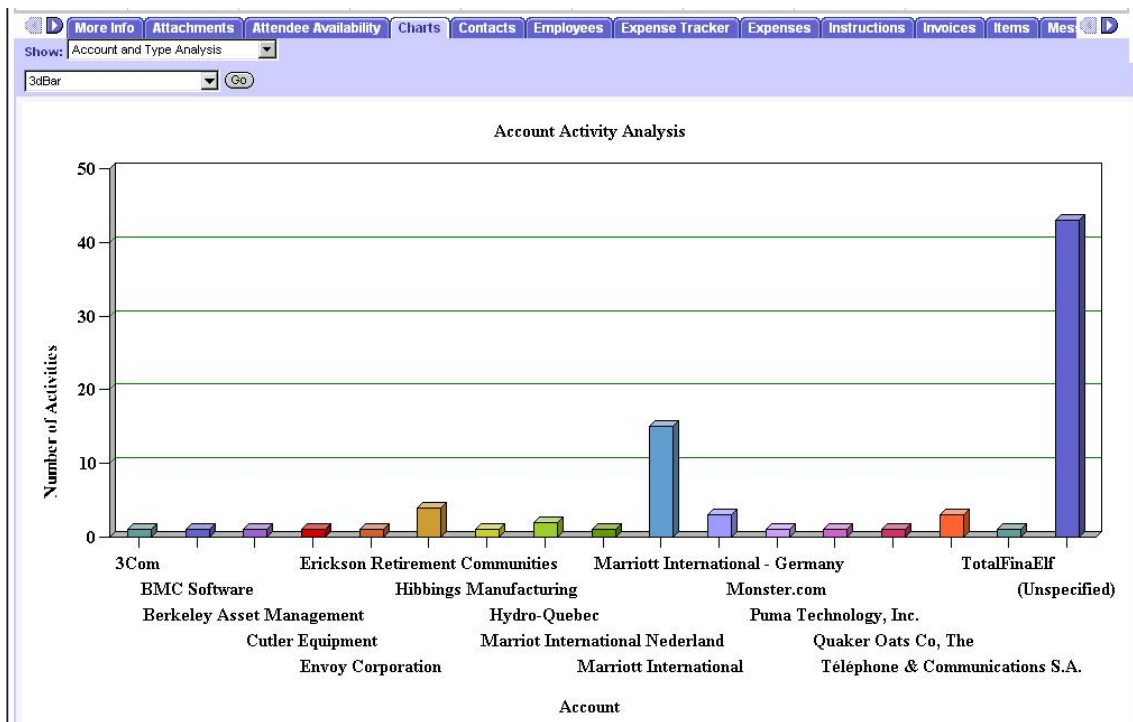


Chart Applet

Association applet. An *association applet* provides the user with the ability to associate records of two business components that have a many-to-many relationship. It is invoked from the check mark icon in a multi-value group applet. An association applet appears below.

Add Contacts - Microsoft Internet Explorer

New Query 22 - 28 of 28+ Query Results Find starting with

| Last Name | First Name | Mr Ms | Account | Job Title | Work Phone # |
|-----------|------------|-------|------------------------|---------------------|----------------------|
| Adams | Al | | Aqua Inc. | | |
| Adams | Betty | | Aqua Inc. | Bus Consint | (613) 555-4707 |
| Adams | Jamie | | Marriott International | VP, Marketing | (310) 455-7664 |
| Aitken | Mary | Ms | Kellogg Company | Program Mgr, Siebel | (616) 555-6344 |
| Akana | Wanda | Mrs. | Bell Canada | Mgr | (514) 555-2504 |
| Aker | Charles | Mr. | Manulife Financial | Inforte | (312) 555-0900 x2202 |
| Alacon | Terry | Ms | Tower Industries | Manager, IT | (415) 555-7834 |

OK Cancel

Association Applet

Explorer or Tree applet. A *tree applet* is used to create an explorer view that allows the user to navigate hierarchically through a structured list of object instances. A tree applet appears below.

Home Accounts Contacts Service Assets Info Center Campaigns Opportunities Quotes Communications Activities Calendar Employee

Show: Explorer

Accounts

- 3Com Corporation
- AMCO Communications
- Acme Inc.
- Alliance Program
- Andercort Consulting
- Andronico
- BG Edwards, Inc
- Ball Systems
- Barracuda
- Berkeley Asset Management
- Berkeley Process Control, Inc.
- Bonanza
- Butler Informatics Consulting

Account:

Accounts 1 - 7 of 56+

| New | Name | Site | Main Phone # | Territories | Industries | Status | URL |
|-----|--------------------|--------------|----------------|-------------|----------------------|--------|------|
| | 3Com Corporation | Headquarters | (415) 329-6500 | | manufacturing indus | Active | |
| | AMCO Communicati | San Rafael | (415) 491-2300 | | steel pipe & tubes | Active | |
| | Acme Inc. | HQ | (510) 245-3000 | | advertising agency | Active | |
| * | Alliance Program | | | | | | |
| | Andercort Consulti | Palo Alto | (415) 389-5400 | | services | Active | |
| | Andronico | Berkeley | (510) 568-1200 | | groceries & related | Active | |
| | BG Edwards, Inc | HQ | (212) 489-1500 | | theatrical producers | Active | www. |

Explorer or Tree Applet

File attachment applet. *File attachment applets* provide access to external documents, such as spreadsheets, word processing documents, and presentations, that have been imported in compressed format into records in a Siebel application. A file attachment applet appears below.

The screenshot shows a Siebel 'Opportunity' applet. The top bar includes 'New' and 'Query' buttons. The main area is divided into several sections:

- Name:** 30 Seats, Siebel Mid-Market - Reve
- Sales Method:** Standard Sales Process
- Committed:** ☐
- *Close Date:** 05/31/2002
- Sales Team:** TSMYTHE
- Sales Stage:** 03 - Qualification
- Revenue:** \$20,000.00
- Organization:** Default Organization
- Territories:**
- Account:** Reverse National Corporation
- Probability %:** 45%
- Lead Quality:** 2-Very High
- Description:** 30 Inside Sales Professionals - really blended agents who also do outbound marketing.
- Site:** HQ
- Expected Value:** \$9,000.00
- Source:**

 Below the main form is a tabbed interface with 'Attachments' selected. It shows a table with columns: Attachment Name, Size (In Bytes), Type, Modified, Local, Download File, Update File, and Comments.

File Attachment Applet

Form Applets

A *form applet* presents business component information in a data entry form layout. An example of a form applet in Siebel Call Center appears in below figure.

The screenshot shows a Siebel 'Incentive Compensation' form applet. The top bar includes 'Calendar', 'Categories', 'Competitors', 'Decision Issues', 'Estimate Compensation', 'Messages', 'Organizations', 'Partners', 'Presentations', and 'Products'. The main area contains the following fields:

- *Name:** Incentive Compensation - 2500 See
- Revenue:** \$2,625,000.00
- Probability %:** 20%
- Committed:** ☒
- Description:**
- Expected Value:** \$525,000.00
- Sales Stage:** 06 - Short List
- Executive Priority:** ☐
- Sales Team:** SADMIN
- Best Case:** \$0.00
- Status:**
- *Close Date:** 11/30/2002
- *Opportunity Currency:** USD
- Worst Case:** \$0.00
- Reason:**
- *Created:** 3/1/2001 9:34:06 AM

Form Applet in Siebel Call Center

List Applets

A *list applet* allows simultaneous display of data from multiple records and presents business component information in a list table format with multicolumn layout with each record of data represented in a row. In addition to textual data, lists also support images in JPEG and GIF formats and edits control such as check boxes, drop-down lists, noneditable MVGs, and text fields.

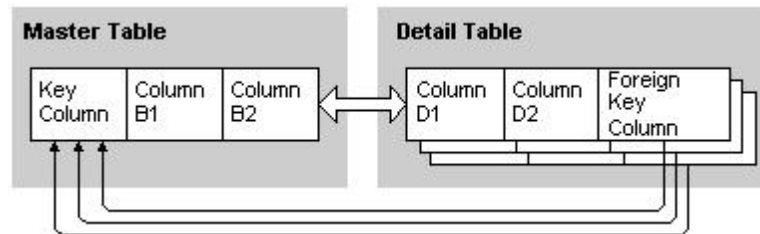
The screenshot shows a Siebel 'Accounts' list applet. The top bar includes 'New', 'Save', and '1 - 7 of 7+'. The table below displays account information:

| New | Name | Site | Main Phone # | Territories | Industries | Status | URL |
|-----|---------------------------|--------------|----------------|-------------|---------------------------------|--------|----------------|
| | 3COM | Santa Clara | (408) 326-5000 | | manufacturing industries | Gold | www.parker.com |
| | 3Com Corporation | Headquarters | (415) 329-6500 | | manufacturing industries | Active | |
| | 3Com Services | US | (415) 329-6500 | | manufactured hardware (general) | Active | |
| | AMCO Communications | San Rafael | (415) 491-2300 | | steel pipe & tubes | Active | |
| | Acme Inc. | HQ | (510) 245-3000 | | advertising agencies | Active | |
| | Acme New England Division | Boston | (617) 232-1121 | | animal specialties | Active | |
| * | Alliance Program | | | | | | |

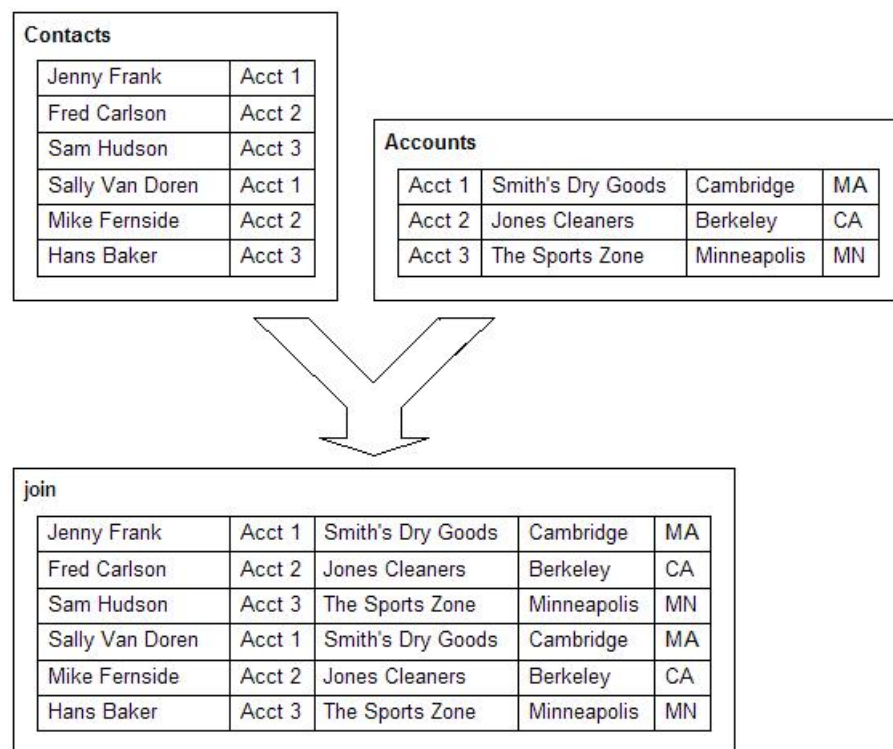
List Applet in Siebel Sales

11. Explain in detail about Joins?

A Join object definition creates a relationship between a business component and a table other than its base table. The join allows the business component to use columns from that table. The join uses a foreign key in the business component to obtain rows on a one-to-one basis from the joined table, even though the two do not necessarily have a one-to-one relationship.



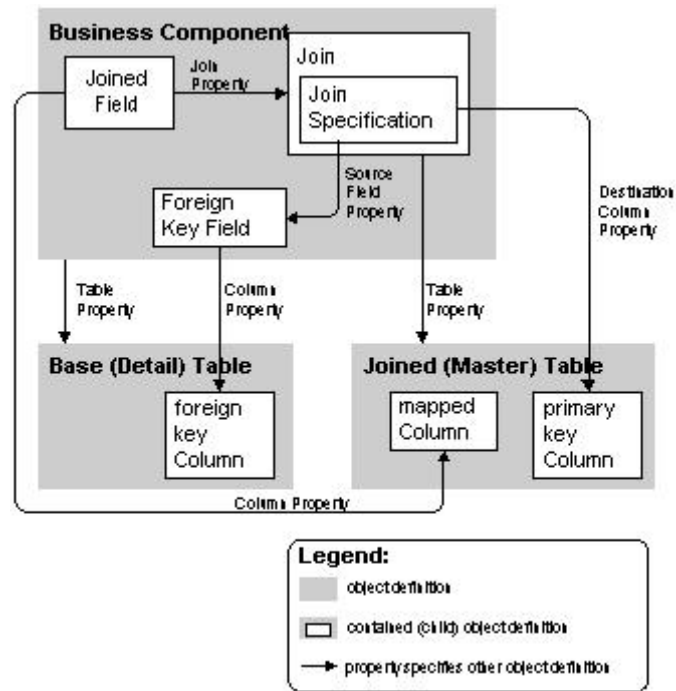
Master-Detail Relationship in a Join



Set of Rows Resulting from a Join

A join is always one-to-one and it is always between a business component and a table. Once a join is created, you can create additional fields in the business component based on columns in the joined table. In the diagram, the account name, city and state are fields that can be added to the Contact business component because of this join.

How a Join is constructed:



Join Relationships

The roles of the object definitions in the diagram are summarized as follows:

Business Component object type. The business component is the parent object definition of the join. Because of the join, fields in the business component (called joined fields) can represent columns from the joined table.

Joined field. A joined field in the business component represents a column from a table other than the business component's base table. Therefore, a joined field must obtain its values through a join. A joined field has the name of the join in its Join property. Together the Join property and Column property identify the column and how to access it. When creating a joined field in a business component, you can change the Type property from the default DTYPE_TEXT to a more appropriate type. For example, if you are joining a table column that contains phone numbers, you can change the Type field to DTYPE_PHONE.

Join object type. Join is a child object type of the Business Component object type. The Join object definition uniquely identifies a join relationship for the parent business component and provides the name of the destination (joined) table. The join object definition identifies the joined table in the Table property. The name of the base table is already known to the business component. Typically, a join object definition is given the same name as the joined table.

Join Specification object type. The join specification object definition is a child of the join object definition. It identifies the foreign key field in the business component and the primary key column in the joined table.

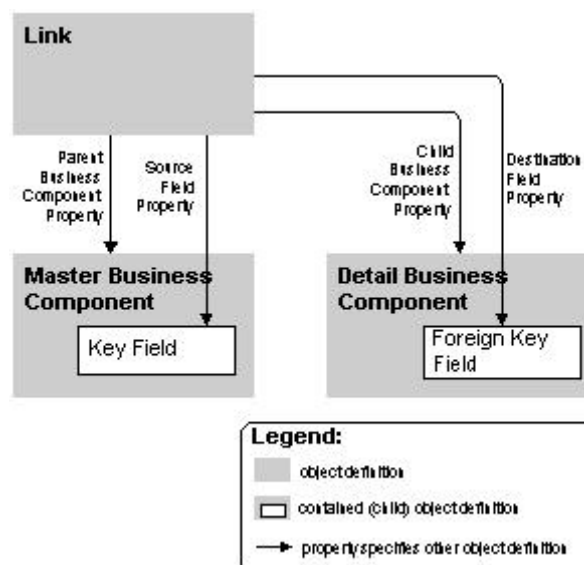
12. Describe Links in detail.

A link implements a one-to-many (or master-detail) relationship between business components based on their base tables. The Link object type makes master-detail views

possible, in which one record of the master business component displays with many detail business component records that correspond to the master.

Link in a Master-Detail View

How a Link is constructed:



Link. The Link object definition specifies a master-detail relationship between two business components. It identifies the master and detail business components, the key field in the master business component, and the foreign key field in the detail business component.

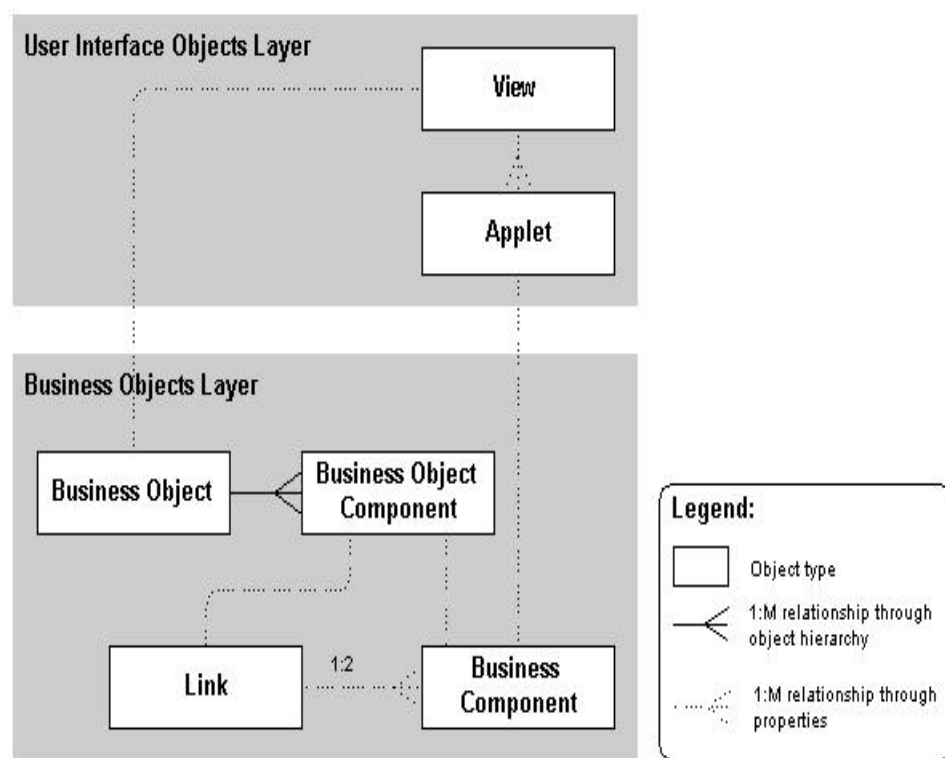
Master business component. The master business component is the “one” in the one-to-many relationship. The name of this object definition is specified in the Parent Business Component property in the Link object definition.

Detail business component. The detail business component is the “many” in the one-to-many relationship. The name of this object definition is specified in the Child Business Component property in the Link object definition.

Source (primary key) field. The source field, also known as the primary key field, is a field in the master business component that uniquely identifies records in the business component. It represents the ROW_ID column from the business component’s base table. The name of this field is specified in the Source Field property in the Link object definition. Source field typically, but not necessarily, represents the row id column from the business component’s base table.

Destination (foreign key) field. The destination field, also known as the foreign key field, is a field in the detail business component that points back to the master record in the business component. Account Id and Opportunity Id are typical foreign key fields.

Link Architecture



In a master-detail view, a Link object definition is incorporated into a business object (by means of a Business Object Component object definition) to establish the master-detail relationship. This relationship applies to any use of the two business components

together within the context of the business object. Each view specifies the business object it uses in its Business Object property. This forces the view to operate as a master-detail view, as specified in the link, without any additional configuration of the view.

A *multi-value link* implements a special use of the Link object type, which is the maintenance by the user of a list of records attached to a control or list column in an applet. The group of attached detail records is called a *multi-value group*.

13. Describe Views in detail.

Views are typically of the following styles:

List-form view. In a list-form view, a list applet and a form applet display data from the same business component. The list applet appears above the form applet. The form applet presents the same information as the currently selected record in the list applet, with a different arrangement that may include more fields.

The screenshot displays a CRM application interface. At the top, there is a navigation bar with tabs: Home, Accounts, Contacts, Households, Employees, Service, Assets, Orders, Campaigns, Opportunities, Quotes, Communications, and SmartS. Below this, a 'Show:' dropdown is set to 'My Accounts', and a 'Queries:' dropdown is set to 'All Accounts'. The main content area is divided into two sections. The top section, titled 'Accounts', shows a list of accounts with columns: New, Name, Site, Main Phone #, Territories, Industries, Status, and URL. The bottom section, titled 'More Info', shows a detailed form for the selected account '3Com'.

| New | Name | Site | Main Phone # | Territories | Industries | Status | URL |
|-----|--------------------|--------------|----------------|-------------|---------------------|------------------|--------------|
| | 3Com | Headquarters | (773) 326-5000 | | manufacturing indus | Gold | www.3com.com |
| | 3Com Distribution | UK | +0283456857 | | management consul | Active | |
| | 3Com Research | US | (415) 329-6500 | | manufactured hard | Active | www.3com.com |
| * | 9 Telecom | France | +33155206242 | | | | |
| | AMCO Communicati | Chicago, IL | (847) 491-2300 | | steel pipe & tubes | Active | www.amco.net |
| * | Acer America, Inc. | San Jose, Ca | (408) 922-2957 | | computer & softwar | Current Customer | www.acer.com |
| * | Air France | France | +33141567800 | | | | |

| | | | |
|---------------|-------------------|---------------|--------------------------|
| *Name: | Current Volume: | Account Type: | Partner: |
| 3Com | \$3,500,000.00 | Customer | <input type="checkbox"/> |
| Site: | Potential Volume: | Status: | Competitor: |
| Headquarters | \$30,000,000.00 | Gold | <input type="checkbox"/> |
| Main Phone #: | Account Team: | Stage: | Reference: |
| | | | |

Master-detail view:

In a *master-detail* view, typically a form applet and a list applet display data from two business components related by a link. The form applet appears above the list applet. The form applet displays one record from the master business component in the master-detail relationship. The list applet displays all of the records from the detail business component that have as their master record the record currently displayed in the form applet.

| Primary | New | Name | Site | Main Phone # | Territories | Industries | Status | URL | Potential Volume |
|---------|-----|------------------------|------------|----------------|-------------|-----------------|--------|------------------|------------------|
| Y | | Marriott International | HQ | (301) 360-3000 | | hotels & motels | Gold | www.marriott.com | \$5,000,000.00 |
| N | * | MCI WorldCom, Inc. | Denver, Co | (303) 390-6400 | | | | www.wcom.com | |

Master-Detail View in a Siebel Application

Thread bar

The *thread bar* is a navigational tool for the user. It provides the means to navigate from view to view among the views previously visited in the current screen.

| New | Name | Site | Main Phone # | Territories | Industries | Status | URL |
|-----|------------------|-------------|--------------|-------------|------------|--------|-----|
| | 3COM | Santa Clara | | | | | |
| | 3Com Corporation | Headquarter | | | | | |
| | 3Com Services | US | | | | | |

Thread Bar in a Siebel Application

Drilldown Behavior in a View

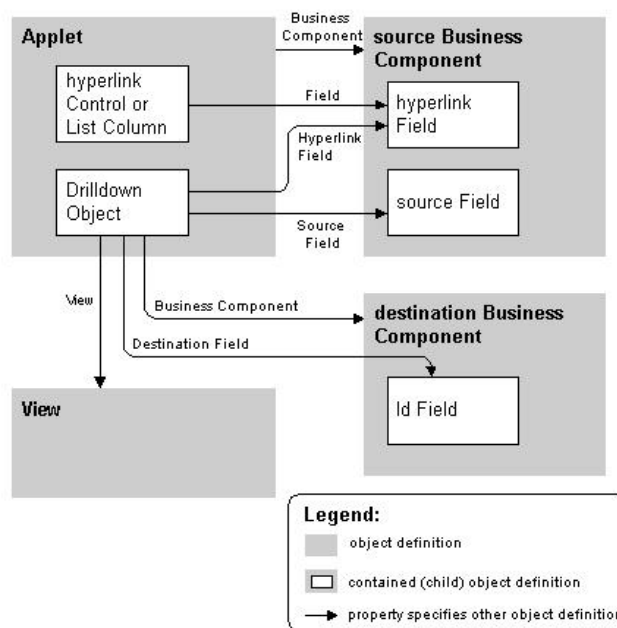
The *Drilldown Object* object type is a child of Applet, used primarily in list applets. It allows the user to drill down from a cell in a list applet (or using a pop-up menu in either a form or list applet) to a particular view. Drilldown controls or list columns in a list applet in Siebel applications consist of colored, underlined text, much like a hypertext link in a Web browser.

| New | Name | Site | Main Phone # | Territories | Industries | Status | URL |
|-----|------|------|--------------|-------------|------------|--------|-----|
|-----|------|------|--------------|-------------|------------|--------|-----|

Drilldown List Columns in a List-Form View

Types of Drilldown Behavior:**1. Static Drilldown Behavior:**

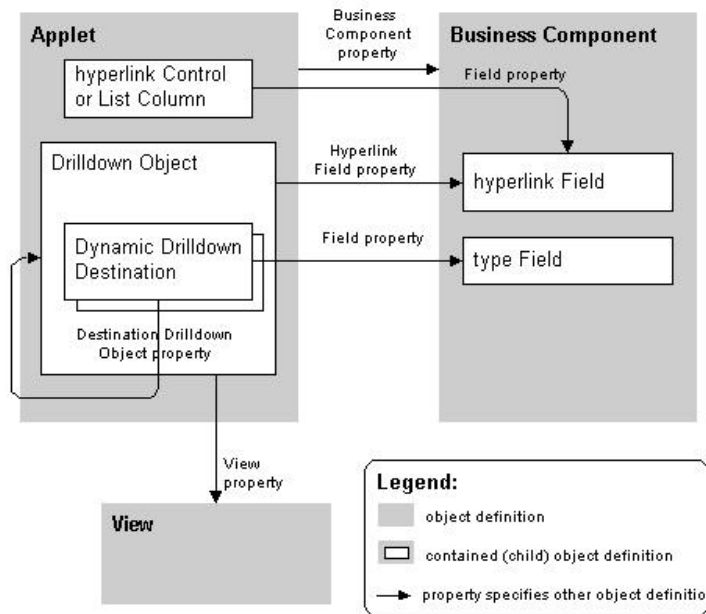
In the example, underlined account name appears in the list column labeled Name. If the user clicks the account in the Name list column, a master-detail view appears, with the selected account in a form applet above an applet displaying the corresponding list of contacts.

**Static Drilldown Configuration**

Dynamic drilldown enables hyperlink navigation to multiple views from the same hyperlink field, depending on the value of a field in the applet's current record.

This is useful in the situation where special processing is desired for various types of contacts, opportunities, accounts, and so on. The business component may have a field that indicates a classification, such as the Lead Quality for an opportunity or the primary Industry for an account. The drilldown behavior can be to check this field in the current record, and navigate to different views for different values found there.

Dynamic drilldown behavior for a hyperlink field (and the corresponding list column or control) is configured with one or more Dynamic Drilldown Destination child object definitions of the Drilldown Object.



Dynamic Drilldown Configuration

14. Describe Screens in detail.

A *screen* is a logical collection of views. It is not a visual construct in itself; rather, it associates views so that other visual constructs, such as the Site Map and tab bar, can reflect the list of views contained in the currently active screen.

A screen does not have a direct relationship with a business object in the same way that a view does. No property in the Screen object type specifies a business object. However, a screen normally contains only views relating to the same business object; this is good design practice. In this sense, it can be loosely said that a screen corresponds to one business object.

The screenshot shows a software application with a menu bar (Home, Accounts, Contacts, Households, Employees, Service, Assets, Orders, Campaigns, Opportunities, Quotes, Communications, SmartS) and a toolbar. The main window displays a list of accounts under the 'Accounts' tab. Below the list is a detailed view of a selected account, showing fields for Name, Site, Main Phone #, Main Fax #, Current Volume, Potential Volume, Account Type, Status, Stage, Expertise, Partner, Competitor, Reference, and PO Approved.

| New | Name | Site | Main Phone # | Territories | Industries | Status | URL |
|-----|--------------------|--------------|----------------|-------------|---------------------|------------------|--------------|
| | 3Com | Headquarters | (773) 326-5000 | | manufacturing indus | Gold | www.3com.com |
| | 3Com Distribution | UK | +0283456857 | | management consul | Active | |
| | 3Com Research | US | (415) 329-6500 | | manufactured hardw | Active | www.3com.com |
| * | 9 Telecom | France | +33155206242 | | | | |
| | AMCO Communicati | Chicago, IL | (847) 491-2300 | | steel pipe & tubes | Active | www.amco.net |
| * | Acer America, Inc. | San Jose, Ca | (408) 922-2957 | | computer & softwar | Current Customer | www.acer.com |
| * | Air France | France | +33141567800 | | | | |

More Info Activities Assets Attachments Contacts ESP Notes Opportunities Profile Revenues Service Profile Service Requests Acc

Name: 3Com **Current Volume:** \$3,500,000.00 **Account Type:** Customer **Partner:** ☐

Site: Headquarters **Potential Volume:** \$30,000,000.00 **Status:** Gold **Competitor:** ☐

Main Phone #: (773) 326-5000 **Account Team:** ADMIN **Stage:** Maintenance **Reference:** ☒

Main Fax #: (773) 329-5555 **Parent:** **Expertise:** **PO Approved:** ☐

Applications of a Screen:

Siebel applications are primarily a collection of screens that users can invoke from the desktop by double-clicking an icon or by pointing a browser to a server running the application. Each combination of screens that is appropriate to a specific class of users can be provided as an application. Siebel Sales, Siebel Service, and Siebel eMarketing are examples of applications. Custom applications can be configured as well, uniquely combining user interface object definitions to meet particular requirements of the organization.

In addition to collecting a group of screens and their views, an application object definition includes the following:

Find object definitions that configure the Find dialog box.

Scripts written in Siebel VB or Siebel eScript and browser JavaScript that can be implemented as event procedures on startup, prior to closing, and so on. These are implemented through Application Script child object definitions, and created and maintained in the Siebel VB or Siebel eScript Editor.

Custom menu options for Siebel-provided methods. These are implemented through the application method menu item object definitions, and created in the Applet Method Menu Item Wizard.

Configuring Application:

When configuring applications for deployment in High Interactivity mode, consider the following:

- Browser scripting is fully supported in High Interactivity mode.
- For fields to interpret and display custom HTML, such as a URL entered by the user, the field's Type property must be set to URL. If it is not set to URL, the HTML is presented and interpreted as plain text. For example, if a user typed a URL in a field of type TEXT, the URL would not be recognized as a link to a Web page.
- You cannot modify the appearance of the rich text editor.
- You cannot modify the background and text color of list applets.
- You cannot place method-invoking controls, such as the delete function, on every row in a list. Instead place a button that calls the method on the applet itself. The function will act on the selected record.

There are cases when an application's configuration file is set to run in High Interactivity mode and all the applets in a view are configured to support High Interactivity, but the view appears in Standard Interactivity mode. Reasons this might occur are:

- One of the applets is in the Query mode. Because High Interactivity implicitly supports query operations from the user interface, it does not support the explicit use of the Query mode.
- One of the applets is in the New mode and uses a New template that is different from the Edit template used in its default mode. This can be avoided by inactivating New templates associated with the applets used in High Interactivity applications. The framework will then default to using the Edit template itself to create new records.
- One of the list applets has multi-row edits or multi-row select enabled.
- One of the list applets is a hierarchical list applet.
- The view uses a template that shows applets in a catalog-style layout. None of the employee applications should be using this layout.
- A combo box picklist uses Long Lists or has an associated pick applet. For example, if you perform an action from a High Interactivity applet that causes a pick applet to be displayed, the pick applet will not be in High Interactivity mode.