

INTRODUCTION

[CUSTOMER RELATIONSHIP MANAGEMENT]

A CRM is a collection of people, processes, software, and internet capabilities that helps an enterprise manage customer relationship effectively and systematically. The goal of CRM is to understand and anticipate the needs of current and potential customer to increase customer retention and loyalty while optimizing the way product and services are sold.

CRM stands for Customer Relationship Management. It is a strategy used to learn more about customers' needs and behaviors in order to develop stronger relationships with them. After all, good customer relationships are at the heart of business success. There are many technological components to CRM, but thinking about CRM in primarily technological terms is a mistake. The more useful way to think about CRM is as a process that will help bring together lots of pieces of information about customers, sales, marketing effectiveness, responsiveness and market trends

The objective is to capture data about every contact a company has with a customer through every channel and store it in the CRM system to enable the company to truly understand customer action. CRM software helps an organization build a database about its customer that management, sales people, customer service provider and even customer can access information to access customer needs with product and offering.

Marketing Automation is the most comprehensive campaign management solution available. It provides everything needed to turn raw, disparate customer data into profitable marketing campaigns – all the way through inception, execution and measurement.

Marketing Optimization applies sophisticated mathematical approaches to optimize marketing campaign ROI given limited budgets, channel capacities and other organizational constraints.

The idea of CRM is that it helps businesses use technology and human resources to gain insight into the behavior of customers and the value of those customers. If it works as hoped, a business can:

- Provide better customer service
- Make call centers more efficient
- Cross sell products more effectively
- Help sales staff close deals faster
- Simplify marketing and sales processes
- Discover new customers
- Increase customer revenues

ARCHITECTURE OF CRM

There are three parts of application architecture of CRM:

1. Operational - automation to the basic business processes (marketing, sales, service)
2. Analytical - support to analyze customer behavior, implements business intelligence alike technology
3. Collaborative - ensures the contact with customers (phone, email, fax, web, sms, post, in person)

Operational CRM

Operational CRM means supporting the so-called "front office" business processes, which include customer contact (sales, marketing and service). Tasks resulting from these processes are forwarded to employees responsible for them, as well as the information necessary for carrying out the tasks and interfaces to back-end applications are being provided and activities with customers are being documented for further reference.

According to Gartner Group, the operational part of CRM typically involves three general areas of business:

- **Sales Force Automation (SFA):** SFA automates some of the company's critical sales and sales force management functions, for example, lead/account management, contact management, quote management, forecasting, sales administration, keeping track of customer preferences, buying habits, and demographics, as well as sales staff performance. SFA tools are designed to improve field sales productivity. Key infrastructure requirements of SFA are mobile synchronization and integrated product configuration.
- **Customer Service and Support (CSS):** CSS automates some service requests, complaints, product returns, and information requests. Traditional internal help desk and traditional inbound call-center support for customer inquiries are now evolved into the "customer interaction center" (CIC), using multiple channels (Web, phone/fax, face-to-face, kiosk, etc). Key infrastructure requirements of CSS include computer telephony integration (CTI) which provides high volume processing capability, and reliability.
- **Enterprise Marketing Automation (EMA):** EMA provides information about the business environment, including competitors, industry trends, and macroenvironmental variables. It is the execution side of campaign and lead management. The intent of EMA applications is to improve marketing campaign efficiencies. Functions include demographic analysis, variable segmentation, and predictive modeling occur on the analytical (Business Intelligence) side.

Analytical CRM

In analytical CRM, data gathered within operational CRM and/or other sources are analyzed to segment customers or to identify potential to enhance client relationship.

Analysis of Customer data may relate to one or more of the following analyses:

1. Contact channel optimization
2. Contact Optimization
3. Customer Segmentation
4. Customer Satisfaction Measurement / Increase
5. Sales Coverage Optimization

6. Pricing Optimization
7. Product Development
8. Program Evaluation

Data collection and analysis is viewed as a continuing and iterative process. Ideally, business decisions are refined over time, based on feedback from earlier analysis and decisions. Therefore, most successful analytical CRM projects take advantage of a data warehouse to provide suitable data.

Collaborative CRM

Collaborative CRM facilitates interactions with customers through all channels (personal, letter, fax, phone, web, e-mail) and supports co-ordination of employee teams and channels. It is a solution that brings people; processes and data together so companies can better serve and retain their customers.

Collaborative CRM provides the following benefits:

- Enables efficient productive customer interactions across all communications channels
- Enables web collaboration to reduce customer service costs
- Integrates call centers enabling multi-channel personal customer interaction
- Integrates view of the customer while interaction at the transaction level

BENEFITS OF CRM SYSTEM

Give customer-facing employees the best tools available

Give your sales and service professionals the chance to deliver stellar customer service every time they are in a call or at a customer site. With optimally configured software solutions from Microsoft Business Solutions, they can find information quickly, answer customer questions satisfactorily, and ensure that your business fulfills or exceeds customer commitments.

Deliver high-quality service at all times

Happy customers generate return business, referrals, and excellent visibility for your business. You want to make doing business with your company efficient and rewarding for them—track customer preferences, be aware of their account histories, and resolve any issues promptly. Microsoft Business Solutions can empower your people to streamline customer communications and interactions, offer timely, reliable information, and demonstrate your organization's accountability.

Boost sales and service performance

Automation of business tasks can go a long way towards streamlining sales and customer service processes, shortening sales cycles, and maintaining consistent quality and productivity on your sales and service teams. In addition, Microsoft Business Solutions offerings allow you to review and enhance the performance of individual sales and service employees as well as entire teams, forecast and plan sales activity, and act on new business opportunities

SALES FORCE AUTOMATION

Sales Force Automation is a technique of using software to automate the business tasks of sales, including order processing, contact management, information sharing, inventory monitoring and control, order tracking, customer management, sales forecast analysis and employee performance evaluation.

SFA is often used interchangeably with CRM; however, CRM does not necessarily imply automation of sales tasks.

Sales Force Automation offers you the most powerful and advanced CRM solution to empower your sales force team. Combined with major e-Marketing features for today's highly competitive e-Business marketplace, this powerful sales automation tool will give your team a strong edge over competitors.

With Sales Force Automation you can get quick responses from your customers, find out their interests and anticipate their future demand for your product and/or services, which will greatly enhance your CRM.

Sales Force Automation is a low cost template-based online solution. Follow our step-by-step guide to complete your program just in a few clicks. Don't wait, start today and experience the power for yourself.

BENEFITS OF SALES FORCE AUTOMATION

Manage all sales opportunities

Immediately create a new opportunity list and turn every lead into a hot prospect. Follow-up using the sales management tool and maximize the number of closed deals.

Forecast revenues accurately

Because the sales information is being continuously updated in real time, everyone on your team, including your outside sales force team will have a clear understanding of future events, and can appropriately forecast product needs and better understand customer demand.

Close more deals

Sales Force Automation will help company enhance the efficiency of sales process and give sales team the tool to collaborate and transform sales prospects into profitable customers.

Interactive and consistent customer leads and deal management

Secure team-wide access to detailed account data will provide easy collaboration among sales force, customer service & support teams, and marketing personnel.

Better understanding of customer needs and long term demands

Analyze current and archived sales data, which can allow company to immediately see changes in customer demands and future needs. Anticipate customers' demands before they happen and be ready for a successful future

OBJECTIVE OF PROJECT

Objectives of the projects are as following-

- i. To manage customer data effectively
- ii. To help the sales people in working easily and efficiently.
- iii. To manage sales person data effectively
- iv. To evaluate sales performance of different sales person.
- v. To generate sales report
- vi. To perform analysis on customer data
- vii. Simplify marketing and sales processes
- viii. To help the sales people in reducing errors that are encountered frequently during manual operations by concurrently updating the data stored in many places.
- ix. To provide security as only an authorized user can interact with the system
- x. To Provide product information to the customer
- xi. Provide a fast mechanism for correcting service deficiencies.
- xii. To provide mechanism to deal with customer complaints effectively.
- xiii. To provide list of customers whenever needed by salesman, administrator or customer relationship executive.
- xiv. To send Offer Letter to customers by finding right customer group.
- xv. To do customer segmentation on basis of income, occupation and territory.
- xvi. To perform analysis of sales data on the basis of product, salesman, and territory.
- xvii. To assign quota of salesman.
- xviii. To assign territory to salesman.
- xix. To provide facility of submitting feedback of the customer.

DRAWBACKS WITH EXISTING SYSTEM

The existing system in the organization is not completely computerized. The system is not working smoothly; therefore the organization has decided to replace it with a completely computerized one. The problems, which the existing system faces, are:

Low Functionality:

With the existing system, the biggest problem was the low functionality of the department. The problem faced hampered the work of the department. For all the tasks like entering the customer data, salesman data, product data, taking the orders, making Bill, making reports etc a large number of employees were appointed who would have been utilized in some other useful tasks.

Erroneous Input And Output:

In the existing system, humans performed all tasks. As in the human tendency, errors are also a possibility. Therefore, the inputs entered by the salesman in the registers may not be absolutely foolproof and may be erroneous. As a result of wrong input, the output reports etc will also be wrong which would in turn affect the performance of home appliance company.

Portability:

System that existed previously was manual. As a result, the system was less portable. One has to carry the loads of so many registers to take the data from one place to another. A big problem was that the system was less flexible and if one wanted to make a change would need to change in all the registers that would also prove to be big headache.

Security:

Security concerns were also one of the motives of the department for the need of the software. In the registers, the data is not secure as anybody can tamper with the data written in the registers. Also for the security of the registers, lots of problems arise to store the registers in a secure place and the appointments of security personnel can also cost a bit.

Whereas in the software, just a password makes it absolutely secure from the reach of unauthorized users.

Data Redundancy:

In the case of manual system, the registers are maintained in which, a lot of data is written. Therefore, there is a problem in the registers that the same data may be repeated again and again. Against the customer id, a lot of data will be repeated which will cause a lot of problems at the time of query as well as at the time of preparing the reports because a single data that will be left mistakenly will largely affect the report and subsequently, the performance of the department. In the software the concept of primary key and foreign key is used very efficiently, which will prevent the redundancy of data that will prove to be very beneficial to the organization because it will nullify the human error completely.

Processing Speed:

In manual system, for a simple work, a number of employees are appointed and in case of keeping records of orders given by the customers, making Bill & reports, they take a lot of time, which may affect the performance of the organization as well as hamper the progress of the organization. It also affects the speed of working in the organization and the work that should have been performed in very short duration can take a large amount of time. But, in the case of software, all the tasks are performed at the touch of a key, which improves the performance of the organization, a great deal, thereby, improving the chances of progress of the organization.

Manual Errors:

When a number of tough tasks are prepared by the humans like preparation of reports, keeping records of all the customers, salesman & available stock in company etc then some human errors are obvious due to a number of factors like mental strain, tiredness etc. But as we all know that computer never gets tired irrespective of the amount of work it has to do. So, this software can nullify the probability of manual errors that improves the company performance.

Complexity In Work:

In a manual system, whenever a record is to be updated or to be deleted a lot of cutting and overwriting needs to be done on all the registers that are concerned with the deleted or updated record, which makes the work very complex. However in the software, once a record is updated or deleted, all the concerned changes are made in the database automatically.

SYSTEM ANALYSIS

IDENTIFICATION OF THE NEED

For identifying the need of the System, we consider the following:

The performance of the System

The performance of the System basically depends on the System Design, tools used in the system, coding, etc. The performance of the system should be fast, accurate & reliable.

The information being supplied and its form

The information being supplied should be very precise, correct and updated. All the information regarding project should be collected from the reliable source. In Project “Customer Relationship Management and Sales Force Automation”, all the information regarding

- Salesman
- Customer
- Product
- Transactions
- Complaint
- Offering
- Schedule
- Customer feedbacks

Should be collected from the Management & Sales People of the organization.

The efficiency of the existing System

The existing System is not well efficient which provokes to develop the new System with some modifications in the older system wherever required. If the existing System is not that much efficient, as required for better results & performance, then it should be converted into the

new advanced & more efficient System. The new System should be well efficient, fast as compared with the older one to show better results than previous one.

The security of the data and software

The Security of both Data & Software is the most important and prime thing in the System Analysis & Design. Data & Software should be kept secured from the unreliable sources otherwise it leads to the Piracy of both data & Software. For proper security username and password is provided to every user of the system. They can access the system as per their authority.

The security of the equipment and personnel, etc

The security of the equipment and personnel, etc is also very important in the System design. Proper measures are to be taken for the security of Equipments, Personnel, etc for the proper functioning of the System.

PRIMARY INVESTIGATION

Evaluation of project request is major purpose of preliminary investigation. It is the collecting information that helps committee members to evaluate merits of the project request and make judgment about the feasibility of the proposed project.

At the heart of any system analysis is detailed understanding of all-important facts of the business area under investigation. The key questions are-

- What is being done?
- How it is being done.
- How frequently does it occur?
- How great is the volume of transactions or decision.
- How well is the task being performed?
- Does a problem exist?
- If a problem exists, how serious is it? What is the underlying cause?

To answer the above questions, system analysts discuss with different category of person to collect facts about their business and their operations.

When the request is made, the first activity the preliminary investigation begins. Preliminary investigation has three parts-

- **Request clarification**
- **Feasibility study**
- **Request approval**

Request Clarification

An information system is intended to meet needs of an organization. Thus the first step is in this phase is to specify these needs and requirements. The next step is to determine the requirements met by the system. Many requests from employees and users in the organizations are not clearly defined. Therefore, it become necessary that project request must examine and clarified properly before considering system investigation.

Information related to different needs of the System can be obtained by different users of the system. This can be done by reviewing different organization's documents such as current method of storing sales data, complaint data etc. By observing the onsite activities the analyst can get close informations related to real system.

FEASIBILITY STUDY

A feasibility study is a preliminary study undertaken before the real work of a project starts to ascertain the likelihood of the project's success. It is an analysis of all possible solutions to a problem and a recommendation on the best solution to use. It involves evaluating how the solution will fit into the corporation. It, for example, can decide whether an order processing be carried out by a new system more efficiently than the previous one.

Technical Feasibility

Technical feasibility refers to the ability of the process to take advantage of the current state of the technology in pursuing further improvement. The technical capability of the personnel as well as the capability of the available technology should be considered. Number of technical issues which are generally realized during the feasibility stage of investigations-

- Present technology, which is in case of this system is manual data management which is highly inefficient and inconsistent,
- Technical capacity of proposed equipments to hold data required using the system.
- Upgradability of system if developed
- Technical guarantee of accuracy, reliability, ease of access and data security.

Economic feasibility

one of the most frequently used technique for evaluating effectiveness of the system is economic analysis. The procedure is to determine the benefits and saving that are expected with the proposed system and compare it with cost. This involves the feasibility of the proposed project to generate economic benefits. The analyst can conduct analysis of economic feasibility by considering following issues-

- The cost to conduct a full systems investigation.
- The cost of hardware and software for class of application being consider.
- The benefits in the form of reduced cost or fewer costly errors.

In case of “Customer Relationship Management and Sales Force Automation System” the company has to maintain large amount of manpower, money and time resources. Huge amount of money can be saved if this system is automated.

Operational Feasibility

This feasibility is related to human organizational and political aspects. This feasibility is carried out by a small group of people who are familiar with information system technique and

understand the parts of the business that are relevant to the project and are skilled in system analysis and design process. Issues involve in testing operational feasibility-

- Support of management
- Acceptability of current business method
- Need of change in existing system
- Involvement of user in planning and development of the project
- Effect of implementation of the proposed system.

Request Approval

All projects that are submitted for evaluation are not feasible. In general, requests that do not pass all feasibility tests are not perused further unless they are modified and resubmitted as new proposal. In some cases, it also happens that a part of a newly developed system is unworkable and the selection committee may decide to combine the workable part of the project with another feasible proposal.

The project development is feasible technically, operationally and economically so it is feasible to develop the project.

SOFTWARE ENGINEERING PARADIGM APPLIED

Software Engineering is the establishment and use of sound engineering principles in order to obtain economically software that is both reliable and works efficiently on real systems.

To solve actual problems in an industry / organizations setting, such as Central or State Govt. Sectors, Public / private Sectors, Colleges, Schools, etc., a Software Engineer or a team of engineers must incorporate a development strategy that encompasses the process, methods, tools layers, and the generic phases. This strategy is referred to as a Process Model or a Software Engineering Paradigm.

Models used for system development in project “Customer Relationship Management and Sales Force Automation” , is Prototyping. Descriptions of these models are as following-

PROTOTYPE MODEL

Prototyping is the process of building a model of a system. In terms of an information system, prototypes are employed to help system designers build an information system that intuitive and easy to manipulate for end users. Prototyping is an iterative process that is part of the analysis phase of the systems development life cycle.

During the requirements determination portion of the systems analysis phase, system analysts gather information about the organization's current procedures and business processes related the proposed information system. In addition, they study the current information system, if there is one, and conduct user interviews and collect documentation. This helps the analysts develop an initial set of system requirements.

Prototyping can augment this process because it converts these basic, yet sometimes intangible, specifications into a tangible but limited working model of the desired information system. The user feedback gained from developing a physical system that the users can touch and see facilitates an evaluative response that the analyst can employ to modify existing requirements as well as developing new ones.

The prototyping model is a software development process that begins with requirements collection, followed by prototyping and user evaluation. Often the end users may not be able to provide a complete set of application objectives, detailed input, processing, or output requirements in the initial stage. After the user evaluation, another prototype will be built based on feedback from users, and again the cycle returns to customer evaluation. The cycle starts by listening to the user, followed by building or revising a mock-up, and letting the user test the mock-up, then back.

Some Advantages of Prototyping:

- Reduces development time.
- Reduces development costs.
- Requires user involvement.
- Developers receive quantifiable user feedback.
- Facilitates system implementation since users know what to expect.
- Results in higher user satisfaction.
- Exposes developers to potential future system enhancements.

SOFTWARE REQUIREMENT SPECIFICATIONS

The Software Requirements Specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of System Engineering are refined by:

- Establishing a complete information description of the System.
- A detailed functional description.
- A representation of System behavior .
- An indication of performance requirements and design constraints.
- Appropriate validation criteria, &
- Other information pertinent to requirements.

Introduction

Purpose

The purpose of the project is to develop a system, which will helpful in management of customer, sales, and complaint data effectively to manage relationship with customer effectively. Proper management of sales data can be used for performing various type of analysis, which can be used for planning marketing strategies and sales planning in effective way.

Product Scope

The software is accessible by different employees of the organization with different access to data. The data should be very secure because as organization is developing this system to manage good customer relation, this data is high confidential in development of business strategy and marketing plans. Data of customer, product and salesman once enter in the database cannot be deleted as other tables are related to them. Complaint should be resolve as per date of complaint and their priority. Performance Evaluation of marketing strategy should be done time-to-time.

Overall Description

Product Functions

- The organization has many salesmen for selling different products.
- Territory is assigned to each salesman and salesman has to sell product to match quota assign to him.
- After each transaction salesman should maintain data of order and customer.
- There is a list of prospective customer, on the basis of that list salesman can choose customers in their territory to plan their tour in effective way.
- The system should enter data of different complaints of customers and give a complaint number to customer. Pending complaint should be displayed by the system time to time so that they can be sorted out on time.
- Inventory should be checked before taking any order and after processing of order inventory should be updated.
- There should be provision to maintain feedback, query of customers.
- System should perform various analysis on customer, sales and complaint data on the basis of product, territory etc.

User Classes and Characteristics

User classes of the system are as following

1. **Administrator**- administrator is responsible for following activities-
 - Assignment of Quota to salesmen
 - Assignment of territory to salesmen
 - Adding and modifying Product Data
 - Adding and modifying salesman data
 - Adding stock to the inventory
 - Performing different analysis on transaction, complaint, sales and other data.
 - Generating different reports
 - Making user account and assigning authorities to them.
2. **Salesman**- Salesman is responsible for following activities

- Order processing
- Maintaining customer data
- Sales call management

3. **Customer Relationship Executive** - he is responsible for following activities

- Complaint management
- Input customer feedback

Operating Environment

The system developed in Visual C++ as front end and Oracle10g as Back end. Crystal 10 Reporting Engine is also required.

Minimum hardware requirements are Operating system should be Windows Xp. Laser printer should be attached to the system.

External Interface Requirements:-

User Interfaces

The System should be Graphical User Interface.

Hardware Interfaces

The system will run on System with Pentium 4 processor, 512 MB RAM, 512 MB Cache and 20 GB hard Disk. It will be connected to Laser Printer.

Software Interfaces

The system will run on Window XP Professional. It is based on Visual C++ and Oracle 10g so Oracle 10g is also required.

SYSTEM DESIGN

Design is the most important part of the development phase for any product or system because design is the place where quality is fostered. Design is the only thing, which accurately translates a customer's requirement in to a finished software product or system. The design step produces a data design, an architectural design, and Interface Design and a Procedural Design.

System specialists often refer to this stage as logical description, in contrast to the process of developing the program software, which is referred to as physical design. The system design describes the data to be input, calculated or stored. Individual data items and calculation procedures are written in detail. The procedures tell how to process the data and produce the output.

REQUIREMENT ANALYSIS

Requirement Analysis is the first technical step in the software engineering process. It is at this point that a general statement of software scope is refined into a concrete specification that becomes a foundation for all the software engineering activities that follow.

Analysis must focus on the informational, functional, and behavioral domains of a problem. To better understand what is required, models are created, the problem is partitioned, and representation that depict the essence of requirements and later, implementation detail, are developed.

REQUIREMENT DETERMINATION

An information system is intended to meet the diverse needs of an organization.

The customer relationship management system and sale automation system is required to contain the following features:

- It should contain all the information about the customer such as name, address, salary, occupation, date of birth, phone number.

- It should maintain product details such as product name, features, price, model number, colors.
- It should maintain salesman data such as salesman name, address, salary, occupation, date of birth, phone number, date of joining, manager name.
- It should also maintain the quota of each salesman.
- It should maintain the transaction of products such as sale of the products, addition of the product in the organization etc.
- It should be able to notify that the product stock is about to finish and reorder is required.
- It should check the performance of the salesman by comparing it with some standards set.
- It should feature the security check so that only the authorized person is allowed to manipulate database as per the level assigned.
- It should be user-friendly software.

FUNCTIONAL REQUIREMENTS

Data should be given in a correct form in order to avoid getting erroneous results. The value of transportation cost and the assignment cost should be known prior to calculate the optimal transportation and assignment cost.

- In order to prepare the budgets the proper and the correct data is to be considered.
- The end users should input correct figures so that the Ratio Analysis may result as per expectations.
- Data should be checked for validity
- Consistency between the information provided in different forms under a scheme should be checked
- Messages should be given for improper input data, and invalid data item should be ignored.

PERFORMANCE REQUIREMENTS

The following characteristics were taken care of in developing the system.

- 1. USER FRIENDLINESS:** The system is easy to learn and understand. A naive user can also use the system effectively, without any difficulty.
- 2. RESPONSE TIME:** Response time of all the operations is kept low. This has been made possible by careful planning.
- 3. ERROR HANDLING:** Response to user errors and undesired situations have been taken care of to ensure that system operates without halting in case of such situation and proper error messages are given to user. Different validations are on different field so that only correct data will be entering in the database.
- 4. SAFETY:** The system is able to avoid catastrophic behavior.
- 5. ROBUSTNESS:** The system recovers from the undesired events without human intervention.
- 6. SECURITY:** The system provides protection of information through the mechanism of password incorporated in it. Therefore only authorized people can access it database. There are three category of user in the system administrator, salesman and customer service executive. All have different authorities of using the system.
- 7. ACCURACY:** The system is highly accurate thus its utility is very high.
- 8. COST ELEMENT:** Servicing a given demand in the system doesn't require a lot of money.

ACCEPTANCE CRITERIA

Before accepting the system, we will demonstrate the system work on existing schemes (designed manually) with actual data as well as we will show through test cases that all the conditions are satisfied. The following acceptance criteria were established for the evaluation of the new system.

MODULES DESCRIPTION

SALESMAN DATA MAINTENANCE MODULE

This module is responsible for the updation of salesman_master database. If any new salesman joins the organization or leave the organization or there is any change in the salesman data, then it can be done with the help of this module. Addition, deletion and updation of salesman data are done by this module.

PRODUCT DATA MAINTENANCE MODULE

This module is responsible for the management of data of different product available in the database. Addition, deletion and updation of the product data is done by this module. At the time of adding product this module also add data in inventory and discount_allowed table.

ORDER PROCESSING MODULE

This module is responsible for taking order by the customer and updation of different tables such as bill_details, transaction , inventory and salesman performance. Whenever any customer place order for any product firstly the salesman checks whether the product is available in the inventory or not. If there is sufficient amount of product in the inventory then a unique bill no. is generated and record is saved in bill_details and transaction table. This module also generates bill.

STOCK PROCESSING MODULE

Time to time updation of stock is necessary for the proper functioning of sales. Stock should be greater than reorder level, if stock is less than reorder level than addition of stock in the inventory is required. Time to time report of stock is generated and administrator and salesmen will get the information that which product is in stock and which one is not. If there is need of adding in the inventory than it can be done with the help of this module.

PRODUCT DATA MAINTENANCE

This module is called whenever any new product is added , updated or deleted from the database. If any new record is added in the database than this module then user have to input details of product, inventory and price details. These details are saved in different table after performing validation checks. If there is any change in the features of the product than it also provide facility of editing. This is also responsible for providing information related to different product on demand.

QUOTA ASSIGNMENT

This module is responsible for assigning quota of different salesman for different products. Quota is assigned for one year. Quota assignment is very important for estimating performance of the company and comparing actual performance with standard performance.

ANALYSIS MODULE

Analysis module is responsible for performing analysis on different sales, customer, product data to analyze sales performance, customer sales behavior and sales trend .

Different modules responsible for performing analysis are

SALES DATA ANALYSIS

This Module is responsible for analyzing sales data on the basis of territory, product and salesman. This module compares quota with sale. Also tells with the help of ratios that whether actual sale is above standards or below.

CUSTOMER DATA ANALYSIS

This module analyzes data of customer on the basis of territory, occupation and income group. This module uses customer master, city details, transaction, bill details and complaint. It gives information related to customer satisfaction about particular product. It compares actual sale with complaint and gives ratio between sale and complaint. High ratio mean low customer satisfaction.

CUSTOMER DATA MAINTENANCE

This module is responsible for management of customer data. This data is accessed and maintained by the salesman. If product is sold to ant new customer than data of new customer can be added in the database. If there is any change in the customer record such as change of address etc then the salesmen can edit their records with the help of this module.

DATABASE DESCRIPTION

Different tables in the database are as following-

Table 1: Customer_master table

This table is used to store different informations related to customers.

Primary Key: cust_id

Field	Data Type	Description
CUST_ID	VARCHAR (10)	Customer identification number
CUST_NAME	VARCHAR (30)	Name of the customer
CUST_ADDRESS	VARCHAR (30)	Address of the customer
CUST_CITY	VARCHAR (20)	City of the customer
CUST_STATE	VARCHAR (5)	State of the customer
CUST_PHONE	NUMERIC (10)	Phone no of the customer
CUST_MOBILE	NUMERIC (10)	Mobile number of the customer
CUST_EMAIL	VARCHAR (40)	Email address of the customer
CUST_DOB	DATETIME	Date of birth of customer
INCOME_GROUP	NUMERIC 2)	Income group of the customer
OCCUPATION	VARCHAR (20)	Occupation of the customer.

Table 2: Quota_Assigned

This table stores how much quota assigned to each salesman.

Foreign key: sl_id, pdt_id

Field	Data Type	Description
SL_ID	VARCHAR (6)	Id of sales person
PDT_ID	VARCHAR (6)	Id of product
QUOTA	INTEGER	Quota assign to salesman

Table 3: Product_master table

This table stores different attributes of the products

Primary key: pdt_id

Field	Data Type	Description
PROD_ID	VARCHAR (6)	Product identification number.
PROD_NAME	VARCHAR (20)	Product name
PROD_BRAND	VARCHAR (20)	Brand name of the product
PROD_TYPE	VARCHAR (10)	Type of the product
PROD_SIZE	VARCHAR (10)	Size of the product
PROD_COLORS_AVAIL	VARCHAR (10)	Colors available
PROD_MODEL_NO	VARCHAR (10)	Model number of the product
PROD_DETAILS	VARCHAR (50)	Details related to product
PROD_GUARANTEE	NUMERIC (2)	Guarantee on product

Table 4: Inventory

This table stores details of inventory of the product.

Foreign key pdt_id

Field	Data Type	Description
PDT_ID	VARCHAR (6)	Product identification number.
STOCK	INT	Stock of the product available
REORDER_LEVEL	INT	Reorder level of product
LAST_UPDATE	DATETIME	Date of last update in inventory

Table 5: Discount_allowed

This table stores details of discount allowed on different product and their price
Foreign key- pdt_id

Field	Data Type	Description
PDT_ID	VARCHAR (6)	Product identification number
PURCHASE_PRICE	INT	Purchase price of the product.
PDT_MRP	INT	Maximum retail price of product
MAX_DISCOUNT	INT	Maximum discount allowed
MIN_DISCOUNT	INT	Minimum discount to offer

Table 7: Current_login

This table stores details of user using system currently.
Foreign_key-> login_id

Field	Data Type	Description
LOGIN_ID	VARCHAR (30)	Login id of the user currently using the system
CATEGORY	VARCHAR (10)	Category of user

Table 8: Salesman_performance

This table stores details related to performance of the salesman.
Foreign key: sl_id, pdt_id

Field	Data Type	Description
SL_ID	VARCHAR (6)	Id of sales person
PDT_ID	VARCHAR (6)	Product Id
MONTH	NUMERIC (2)	Month of the sale
YEAR	NUMERIC (4)	Year of the sale
SALE_QUANTITY	NUMERIC (4)	Quantity of the sale

Table 9: Salesman_master

This table stores different informations related to salesman.

Primary key : sl_id

Field	Data Type	Description
SL_ID	VARCHAR (6)	Sales person identification number
SL_NAME	VARCHAR (20)	Sales person name
SL_ADDRESS	VARCHAR (40)	Address of the sales person
SL_CITY	VARCHAR (30)	City of the sales person
SL_PHONE	NUMERIC (10)	Phone number of the sales person
SL_MOBILE	VARCHAR (10)	Mobile number of the sales person
SL_EMAIL	VARCHAR (40)	Email address of the sales person
SL_DOB	DATETIME	Date of birth of the sales person
SL_DATE_JOIN	DATETIME	Date of joining of sales person
MANAGER	VARCHAR (6)	Manage of the sales person
TERRITORY	NUMERIC (20)	Territory assigned to sales person

Table 10: Income_group

This table stores information related to different income groups of the customers.

Primary_key-> group_id

Field	Data Type	Description
GROUP_ID	VARCHAR(2)	Income group id
MIN_INCOME	VARCHAR(7)	Minimum Income
MAX_INCOME	VARCHAR(7)	Maximum income

Table 11: Bill details

Primary key:: bill_no

Foreign key:: cust_id , sl_id

Field	Data Type	Description
BILL_NO	NUMERIC (10)	Bill number
CUST_ID	VARCHAR(6)	Customer Id
SL_ID	VARCHAR (6)	Sales person ID who has made the sale.
BILL_DATE	DATETIME	Date of bill
TOTAL	NUMERIC (10)	Amount of the bill

Table 12: Transaction Table

This table stores details of different transactions.

Foreign key:: bill_no, pdt_id

Field	Data Type	Description
BILL_NO	NUMERIC (10)	Bill number
PDT_ID	VARCHAR (6)	Product id
QUANTITY	NUMERIC (4)	Quantity of product sold

Table 13: Login table

This table is used to maintain data of login id of all users of the system.

Primary key :: login_id

Field	Data Type	Description
LOGIN_ID	VARCHAR (30)	Login id of the user
PASSWORD	VARCHAR (30)	Password
CATEGORY	CHAR (10)	Category of the user

Table 14: Complaint table

This table stores details of complains by customers.

Primary key:: cpl_id

Foreign key:: cust_id, pdt_id

Field	Data Type	Description
CPL_ID	VARCHAR (6)	Complaint id
PDT_ID	VARCHAR (6)	Product id
CUST_ID	VARCHAR (6)	Customer id
DESCRIPTION	VARCHAR (50)	Complaint Description
NATURE	VARCHAR (30)	Nature of the problem
COMPLEXITY	NUMERIC (2)	Complexity of the problem
CPL_DATE	DATETIME	Complaint date
DATE_ACTION	DATETIME	Date of the solution of the complaint
TECHNICIAN	VARCHAR (30)	Name of Technician who has solve the Complaint.
ACTION_TAKEN	VARCHAR (30)	Action taken to solve the complaint
PAYMENT_RECIEVED	NUMERIC (5)	Payment by customer

Table 15: City_details

This table stores details of different cities and territories

Field	Data Type	Description
CITY_ID	VARCHAR (5)	Id of city
CITY_NAME	VARCHAR (30)	Name of the city
STATE	VARCHAR (20)	State of the city
PIN_CODE	VARCHAR (10)	Pin code of the city
CITY_TERRITORY	VARCHAR (10)	Territory of the city

Table 17: primamy_key

This table stores last primary key of different tables. This table consists of only one record and this is updated after insertion of records in different table.

Field	Data Type	Description
SL_ID	VARCHAR(7)	Last id of salesman Id
CUST_ID	VARCHAR(6)	Last id of customer
COMP_ID	VARCHAR (6)	Last id of complaint
PROD_ID	VARCHAR(6)	Last id of product
BILL_NO	VARCHAR(10)	Last bill number

Table 18: Offer_letter table

This table stores details related to promotional offers

Primary key- letter_no

Foreign key: pdt_id

Field	Data Type	Description
LETTER_NO	VARCHAR (10)	Letter number
PDT_ID	VARCHAR (6)	Id of Product
TARGET_GROUP	VARCHAR (20)	Target group of offer
DATE_VALIDITY	DATETIME	Date of validity of letter
SUMMARY	VARCHAR (50)	Summary of letter
OFFER_DESCRIPTION	VARCHAR (200)	Description of offer

HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirement (minimum): -

Processor	:	Pentium IV or above
Hard Disk	:	20 GB
RAM	:	512 MB
Monitor	:	Color VGA
Printer	:	Laser

Software Requirement (minimum): -

Platform / Operating System	:	Windows Xp and above
Front End. / GUI Tools	:	Visual Studio (Visual C++)
RDBMS	:	Oracle 10g
Reporting Tool	:	Crystal Report 10

FRONT END- VISUAL C++

Java supports the writing of many different kinds of executables: applications, applets, and servlets. The basic elements of a Java application are introduced in this chapter. The old adage that practice makes perfect is certainly true when learning a programming language. To encourage programming on the computer, the mechanics of compiling and running a Java application are outlined.

Reason for choosing Java as Front end tool are as following-

Platform Independence

- The Write-Once-Run-Anywhere ideal has not been achieved (tuning for different platforms usually required), but closer than with other languages.

Object Oriented

- Object oriented throughout - no coding outside of class definitions, including main().
- An extensive class library available in the core language packages.

Compiler/Interpreter Combo

- Code is compiled to bytecodes that are interpreted by a Java virtual machines (JVM) .
- This provides portability to any machine for which a virtual machine has been written.
- The two steps of compilation and interpretation allow for extensive code checking and improved security.

Automatic Memory Management

- Automatic garbage collection - memory management handled by JVM.

Security

- No memory pointers
- Programs runs inside the virtual machine sandbox.

- Array index limit checking

Dynamic Binding

- The linking of data and methods to where they are located, is done at run-time.
- New classes can be loaded while a program is running. Linking is done on the fly.
- Even if libraries are recompiled, there is no need to recompile code that uses classes in those libraries.

Good Performance

- Interpretation of byte codes slowed performance in early versions, but advanced virtual machines with adaptive and just-in-time compilation and other techniques now typically provide performance up to 50% to 100% the speed of C++ programs.

Threading

- Lightweight processes, called threads, can easily be spun off to perform multiprocessing.
- Can take advantage of multiprocessors where available
- Great for multimedia displays.
- Java was designed with networking in mind and comes with many classes to develop sophisticated Internet communications.

About Crystal Reports

Crystal Reports is designed to work with your database to help you analyze and interpret important information. Crystal Reports makes it easy to create simple reports; it also has the comprehensive tools you need to produce complex or specialized reports.

Create any report you can imagine

Crystal Reports is designed to produce the report you want from virtually any data source. Built-in report wizards and experts guide you step by step through building reports and completing common reporting tasks. Formulas, cross-tabs, subreports, and conditional formatting help make sense of data and help you uncover important relationships that might otherwise be hidden. Geographic maps and charts communicate information visually when words and numbers are simply not enough.

Extend reporting to the Web

The flexibility of Crystal Reports doesn't end with creating reports—your reports can be published in a variety of formats including Microsoft® Word and Excel, email, and even over the Web. Advanced Web reporting lets other members of your workgroup view and update shared reports inside their web browser.

Incorporate reports into applications

Application and web developers can save time and meet the needs of their users by integrating the report processing power of Crystal Reports into their database applications. Support for most popular development languages makes it easy to add reporting to any application.

Visual C++, Crystal Compatibility:

Whether it's the web master in IT, the promotion manager in marketing, the database administrator in finance, or the CEO, Crystal Reports is a powerful tool designed to help everyone analyze and interpret the information that's important to them. The Crystal Report Designer Component Automation Server (Craxdrt.dll, often referred to as the RDC or Report Designer Component) is designed to take advantage of several features of the Microsoft Visual Basic IDE. However, the RDC can be integrated into other developer tools such as Microsoft Visual C++.

Visual C++ 5 and 6 provide native COM support which the RDC requires. Accessing and implementing this support in Visual C++ is somewhat more involved than is the case with Visual Basic. The method used to expose.

The RDC's object model in Visual C++ also differs as to how it is exposed in Visual Basic. The RDC is broken into two main components. The Automation server component (Craxdrt.dll), and the Crystal Report Viewer control (Crviewer.dll)

The "Previewing a Report" section of this paper shows how to use these two components of the RDC to preview a report. The "Setting report properties" section extends on this and shows how to pass values (such as database logon information or parameter field values) to a report at runtime.

BACK END: ORACLE 10g

Object is an object-relational database. A relational database is an extremely simple way of thinking about and managing the data used in a business. It is nothing more than a collection of tables of data. We all encounter tables every day: weather reports, stock charts, sports scores. These are all tables, with column heading and rows of information simply represented. Even so the relational approach can be sophisticated and powerful enough for even the most complex of business. An object-relational database supports all of the features of a relational database while also supporting object oriented concepts and features.

Unfortunately, the very people who can benefit most from a relational database – the business users – usually understand it the least. Application developers, who must build systems that these users need to do their jobs, often find relational concept difficult to explain in simple term.

Everyone has “Data “

Information philosopher likes to say that data is just data until it is organized in a meaningful way, at which point it becomes “information”. If this is true, then Oracle is also a means of easily turning data into information. Oracle will sort through and manipulate data to reveal piece of knowledge hidden there- such as total buying trends, or other relationship- which are as yet undiscovered. The main point here is that you have data, and you do three basic things with it: acquire it store it, and retrieve it.

Once you have achieved the basics, you can make computation with data, move it from one place to another, or modify it. This is called processing, and, fundamentally, it involves the same three steps that effect how information is organized.

A relational database management system (often called as RDBMS) such as oracle gives you the way of doing these task in an understandable and reasonably uncomplicated way. Oracle basically does three things:

- ❖ Lets you put data into it
- ❖ Keeps the data
- ❖ Let you get the data out and work with it.

.

Tools of Oracle

Oracle is a full-featured relational database management system (RDBMS) that offers a variety of administrative tools to ease the burdens of database development, maintenance and administration. Most frequently used tools are Enterprise Manager, , SQL, Service Manager, Data Transformation Services and.

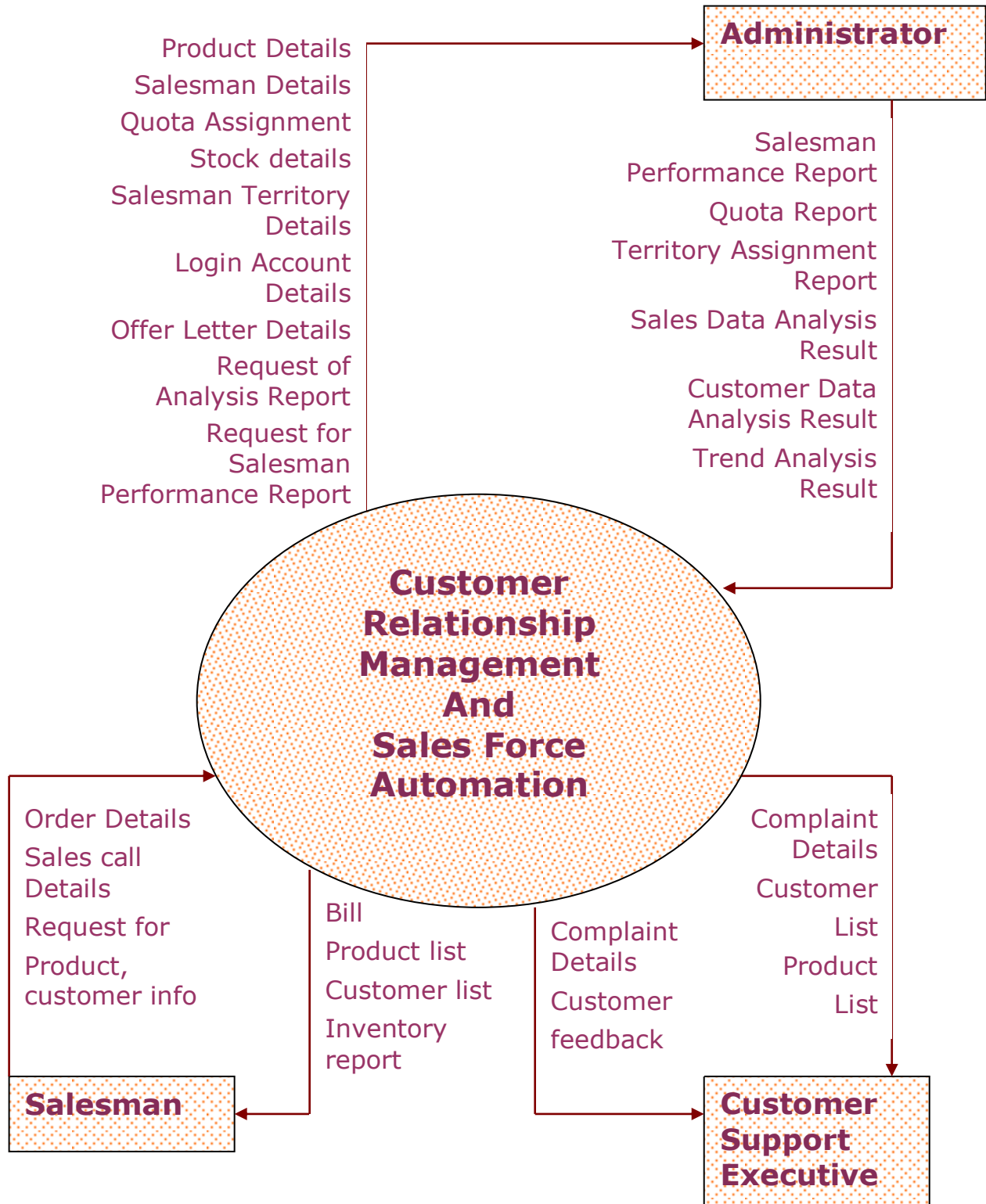
Enterprise Manager is the main administrative console for Oracle installations. It provides you with a graphical "birds-eye" view of all of the Oracle installations on your network.

SQL PLUS offers a quick and dirty method for performing queries against any of your Oracle databases

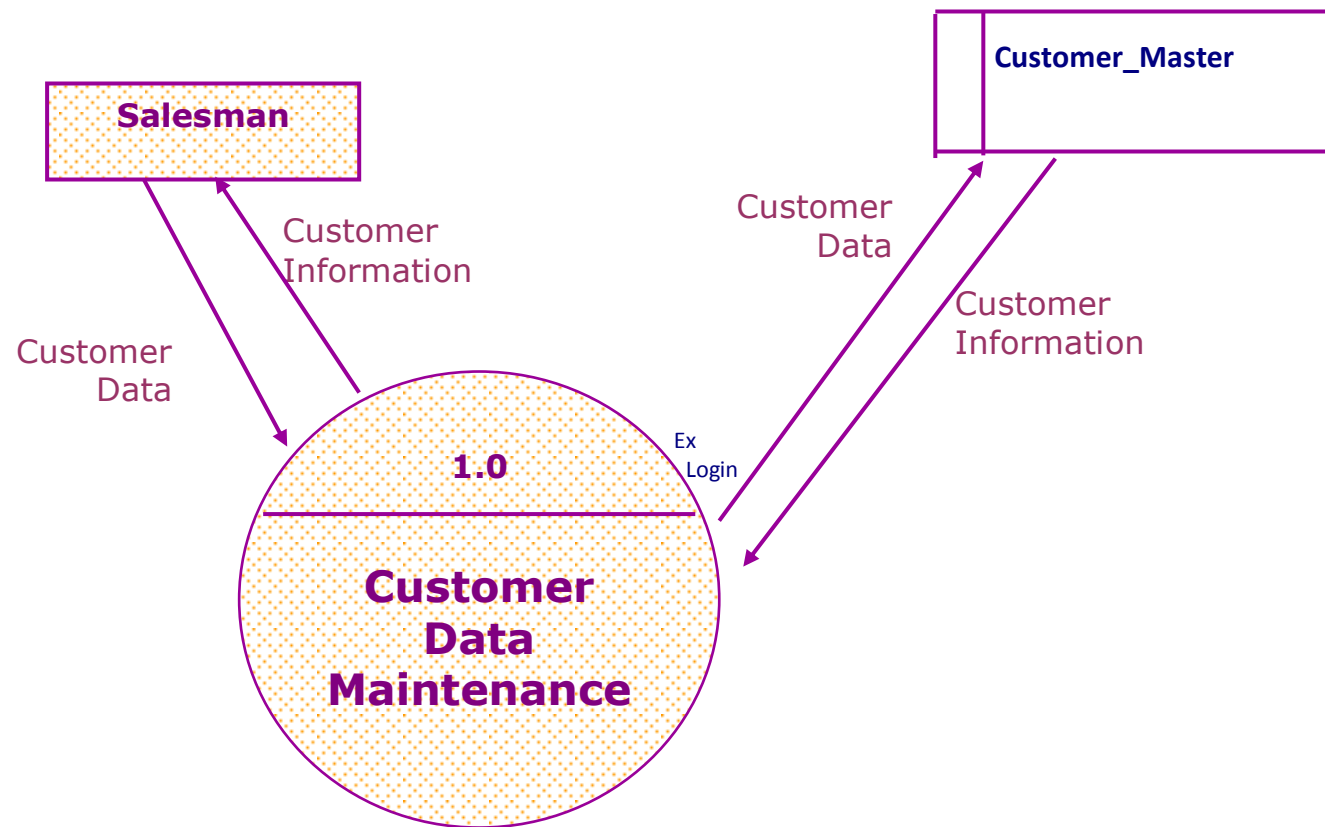
ORACLE Capacities:

Following list shows the system lists for Various ORACLE Objects.

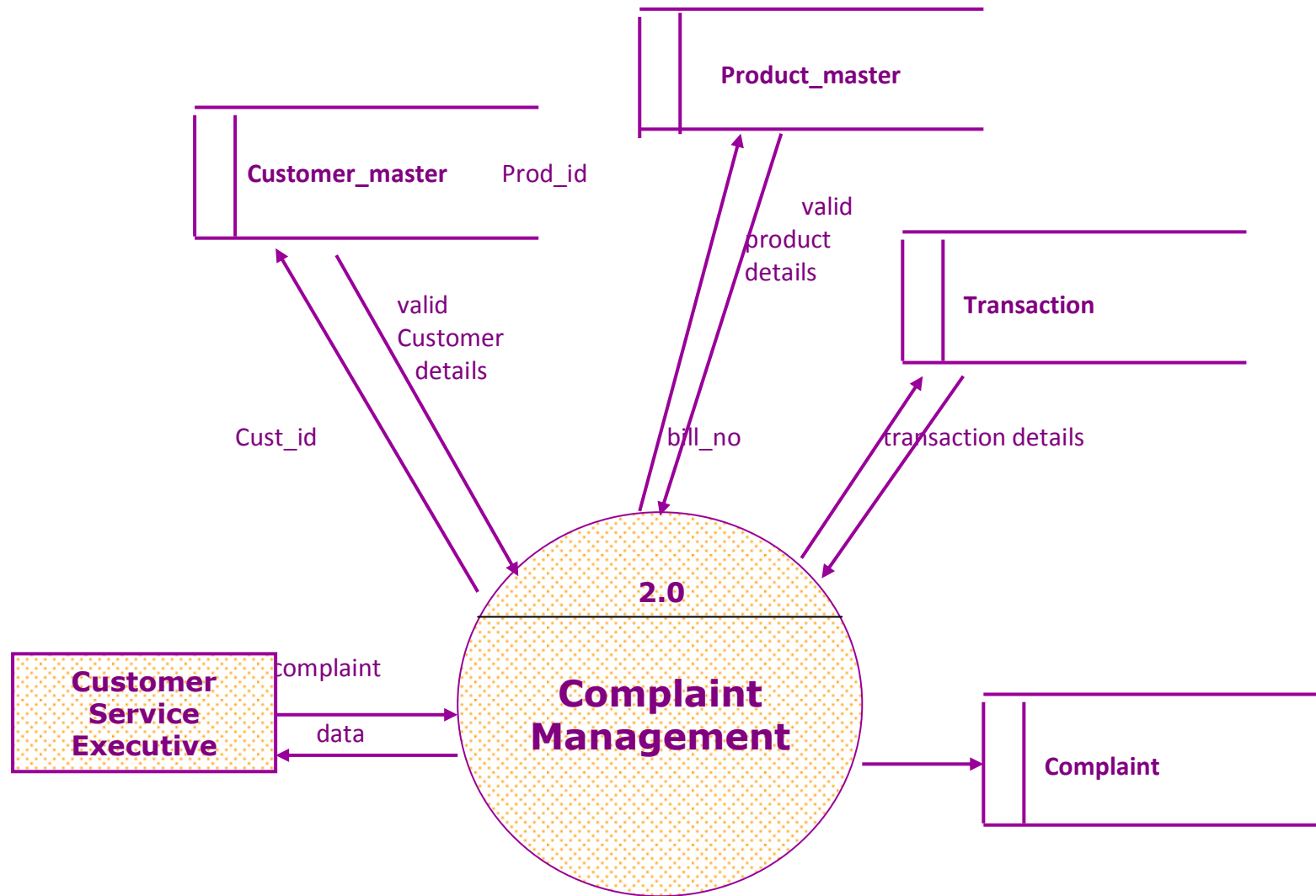
OBJECT	LIMITS
Databases	32,767 databases
Tables	2 billion tables per database
Columns	1024 columns per table
Triggers	Up to 2 billion triggers
User Connections	32767 connections are possible
Open Objects	2 billion open objects at a time
Stored procedures	can contain 255 parameters
Locks	2 billion locks



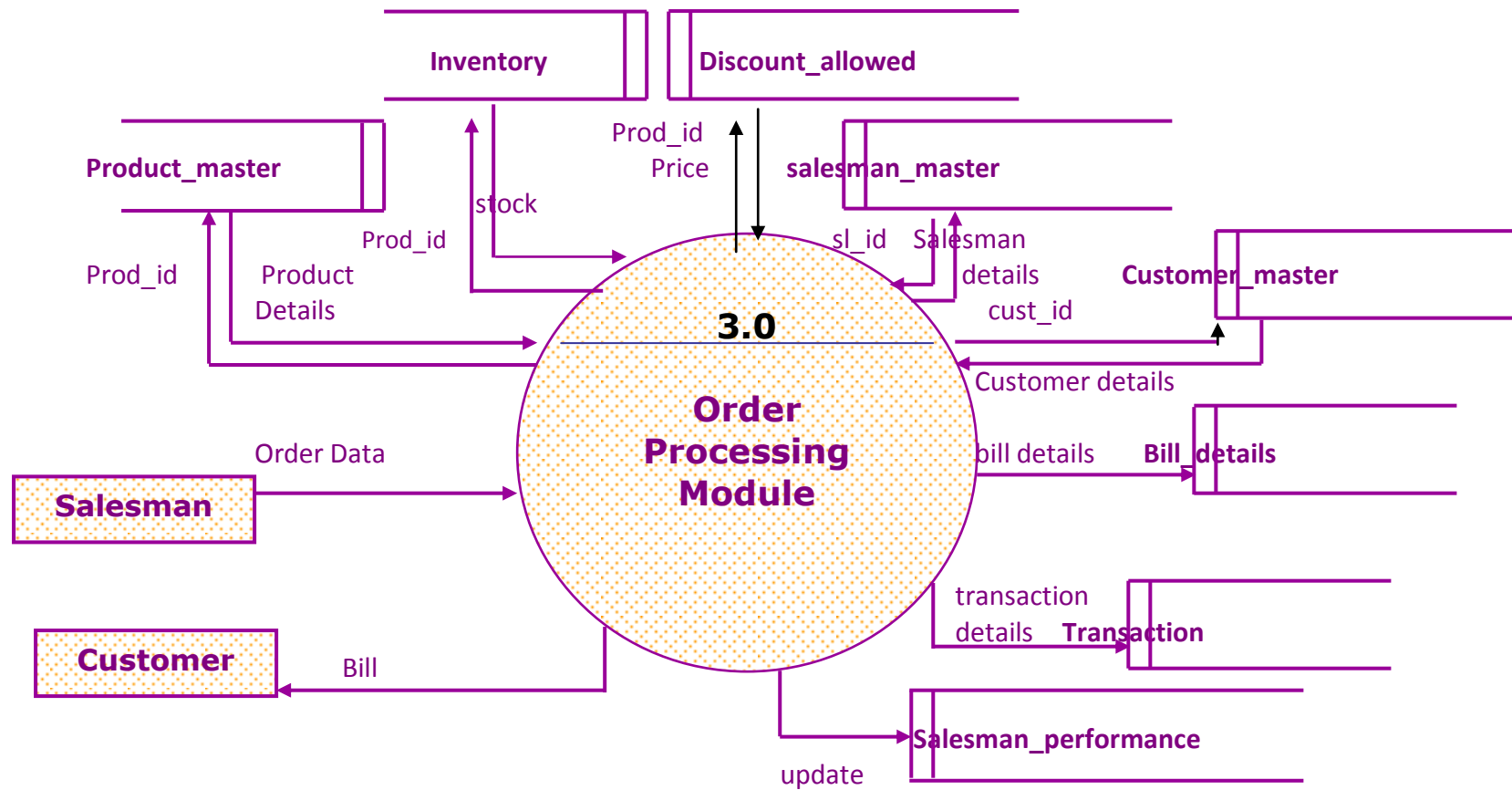
Context Level DFD Of Customer Relationship Management and Sales Force Automation



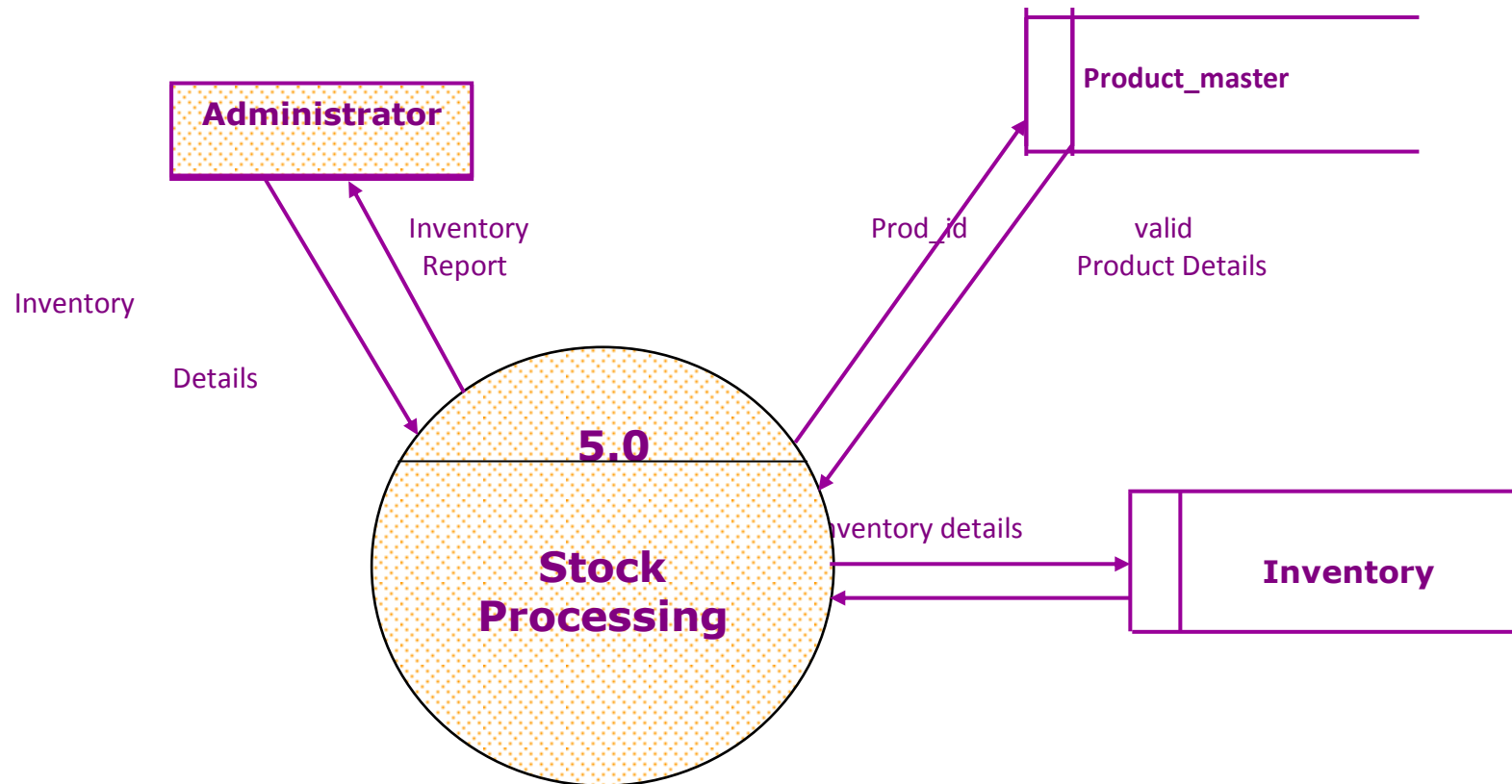
Level 1 DFD of Customer Data Maintenance



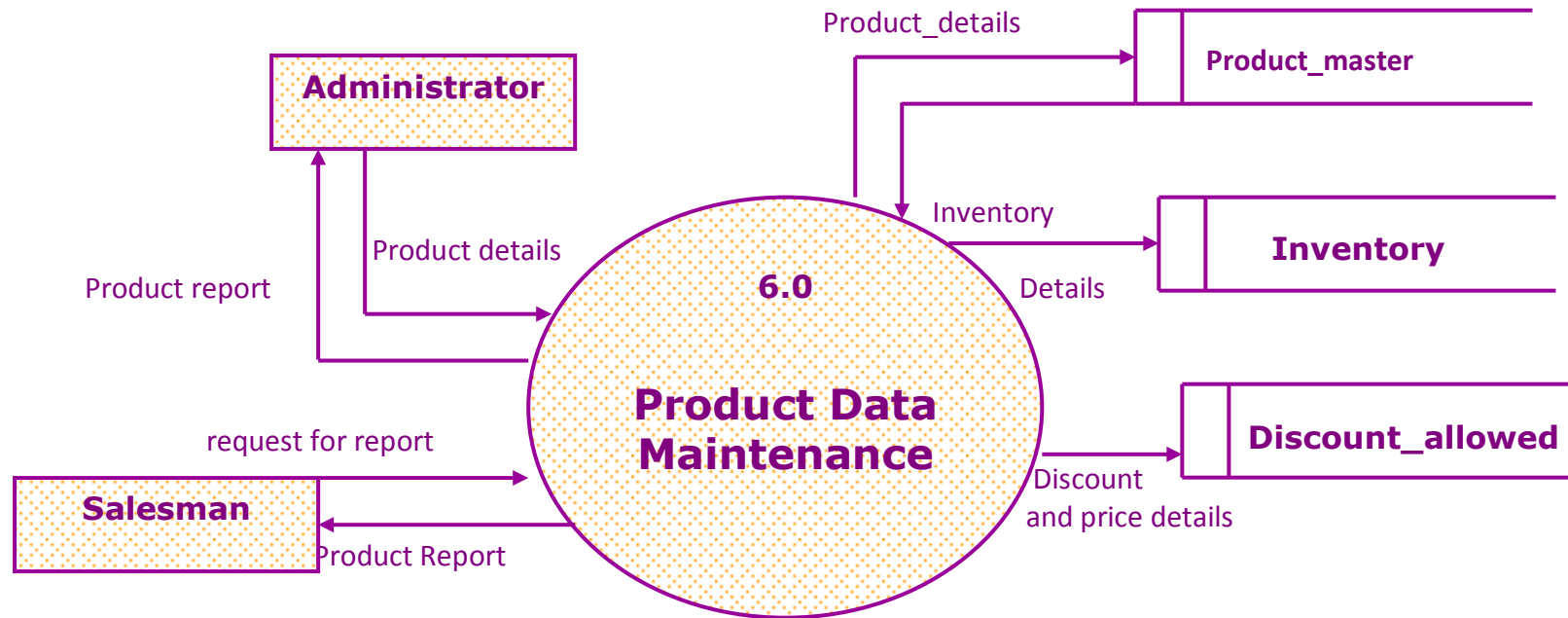
1st Level DFD of Complaint Management



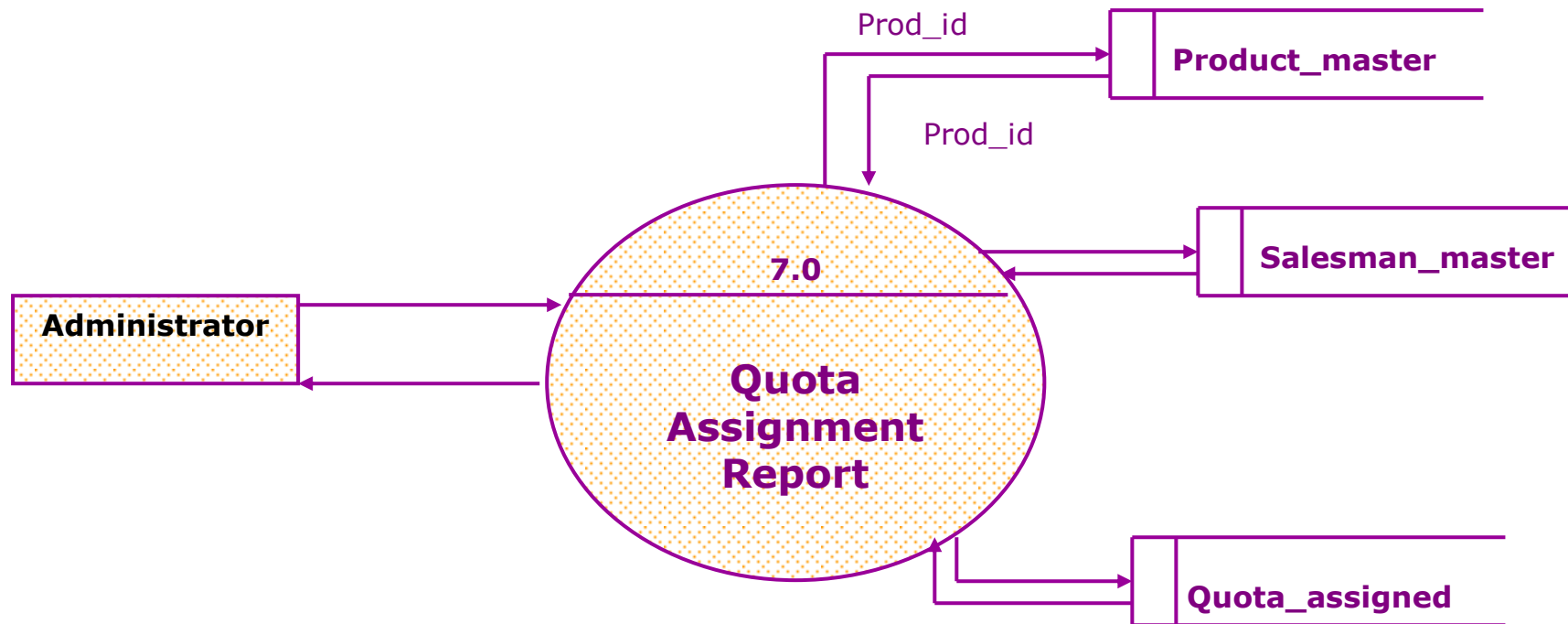
1 level DFD of Order Processing Module



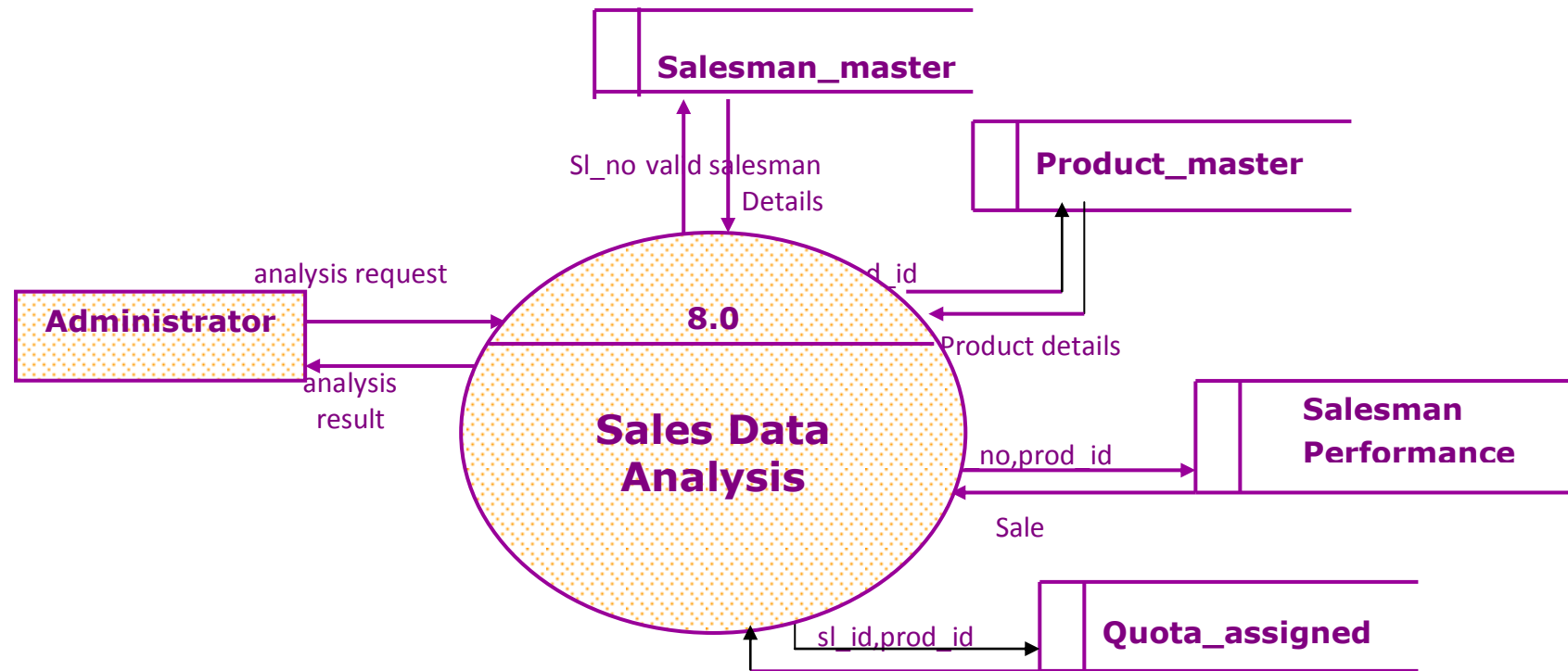
1st Level DFD of Stock Processing Module



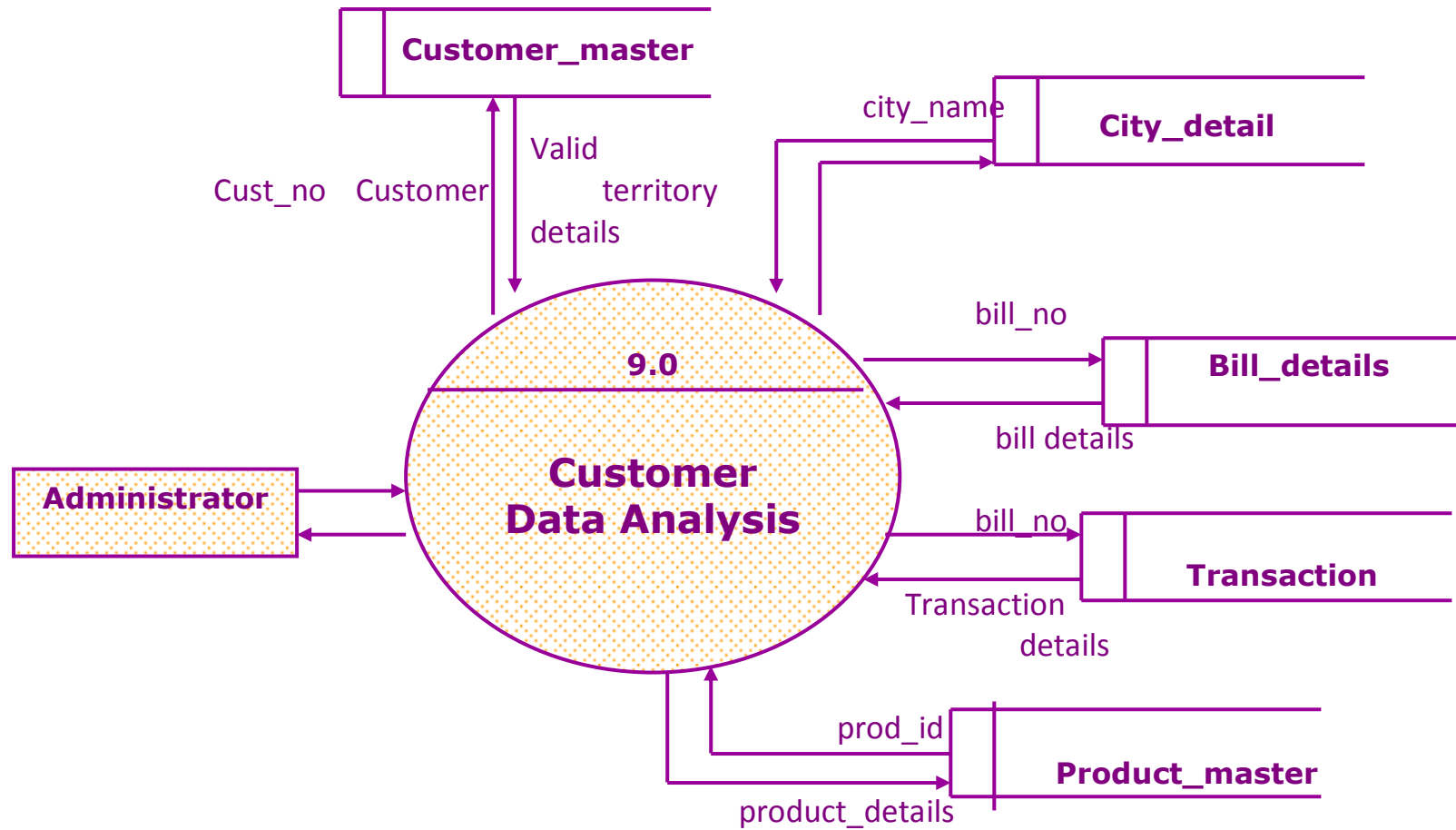
1st Level DFD of Product Data Maintenance Module



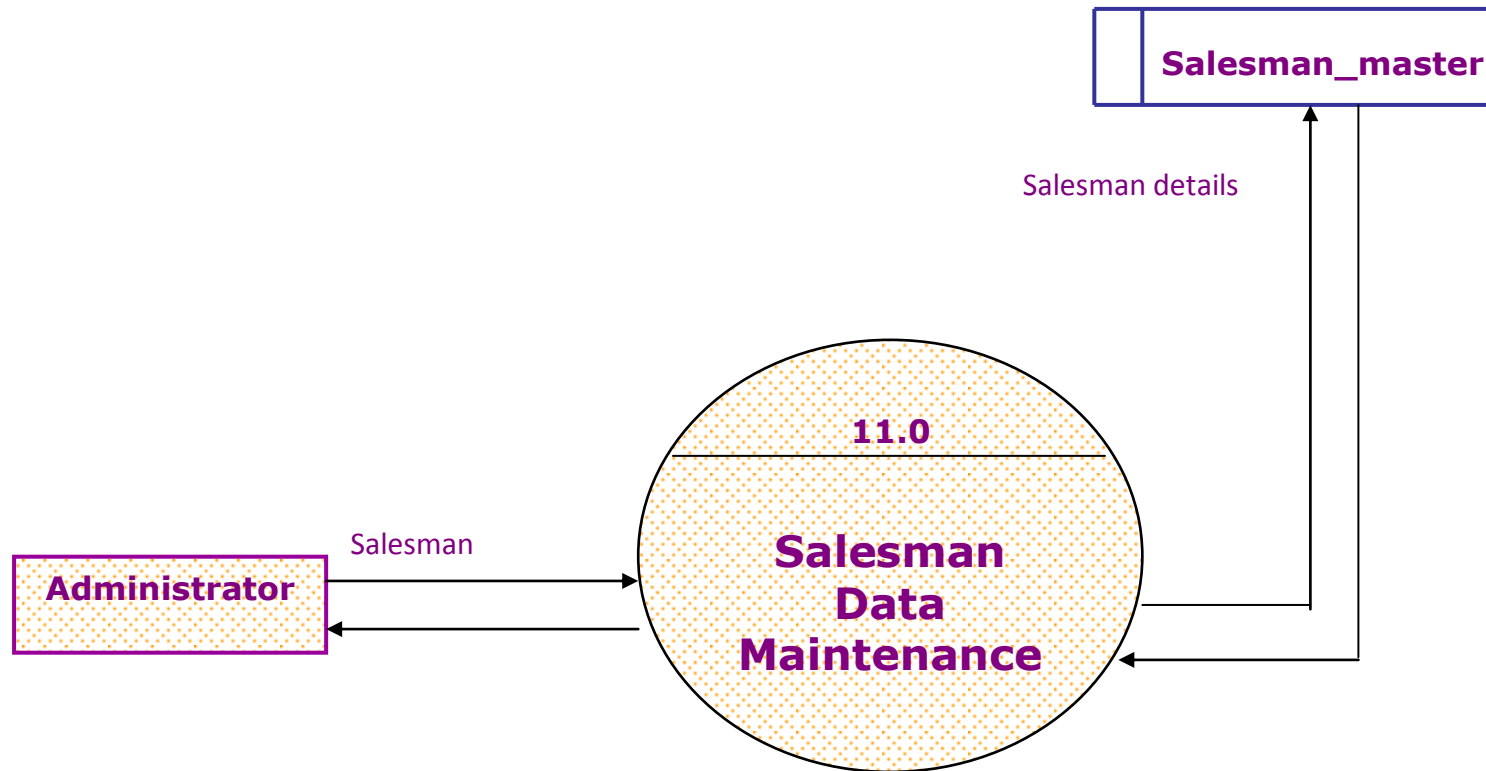
1st Level DFD of Quota Assignment Module



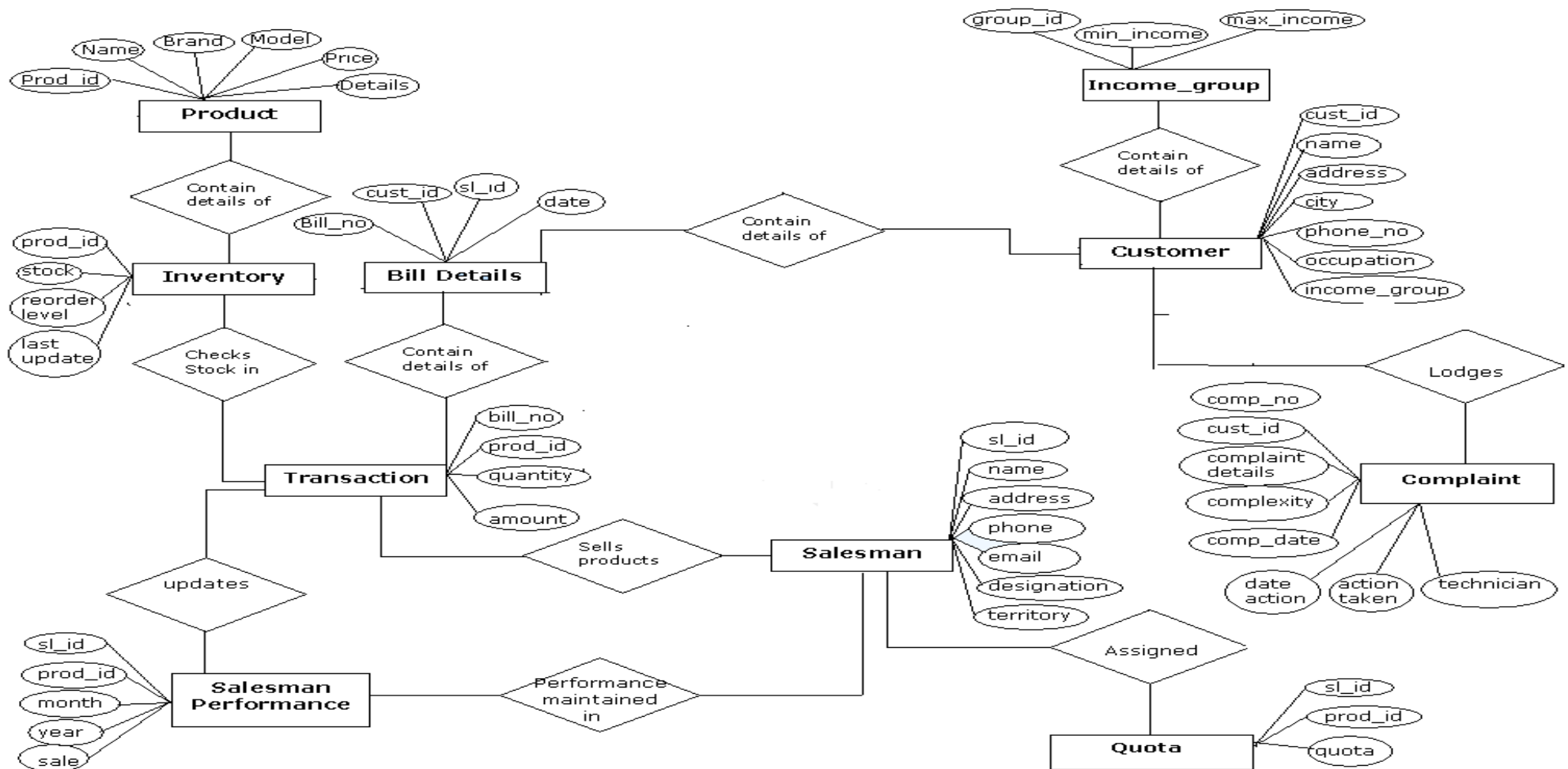
1st Level DFD of Sales Data Analysis Report



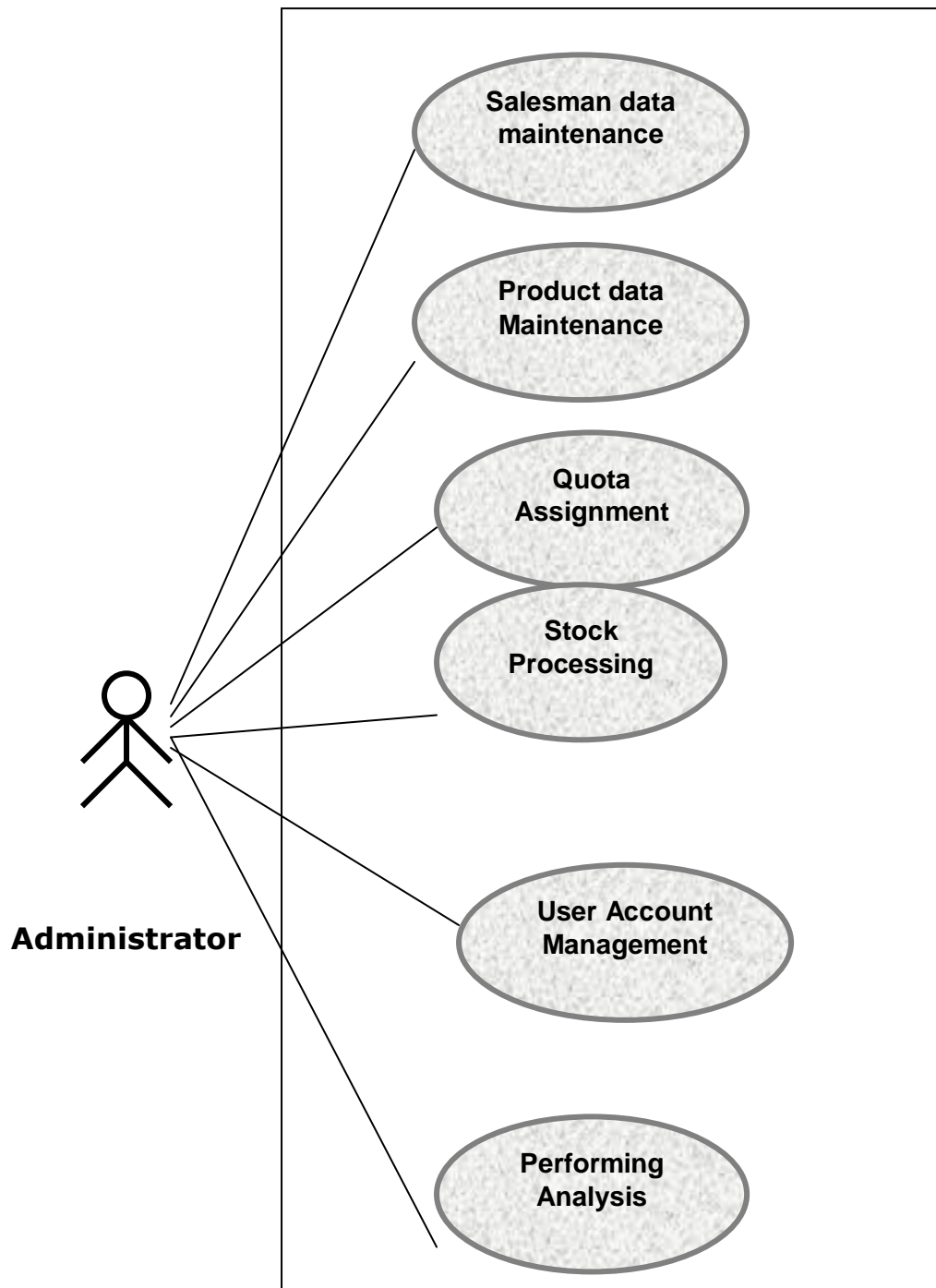
1st level DFD of Customer Data Analysis Module

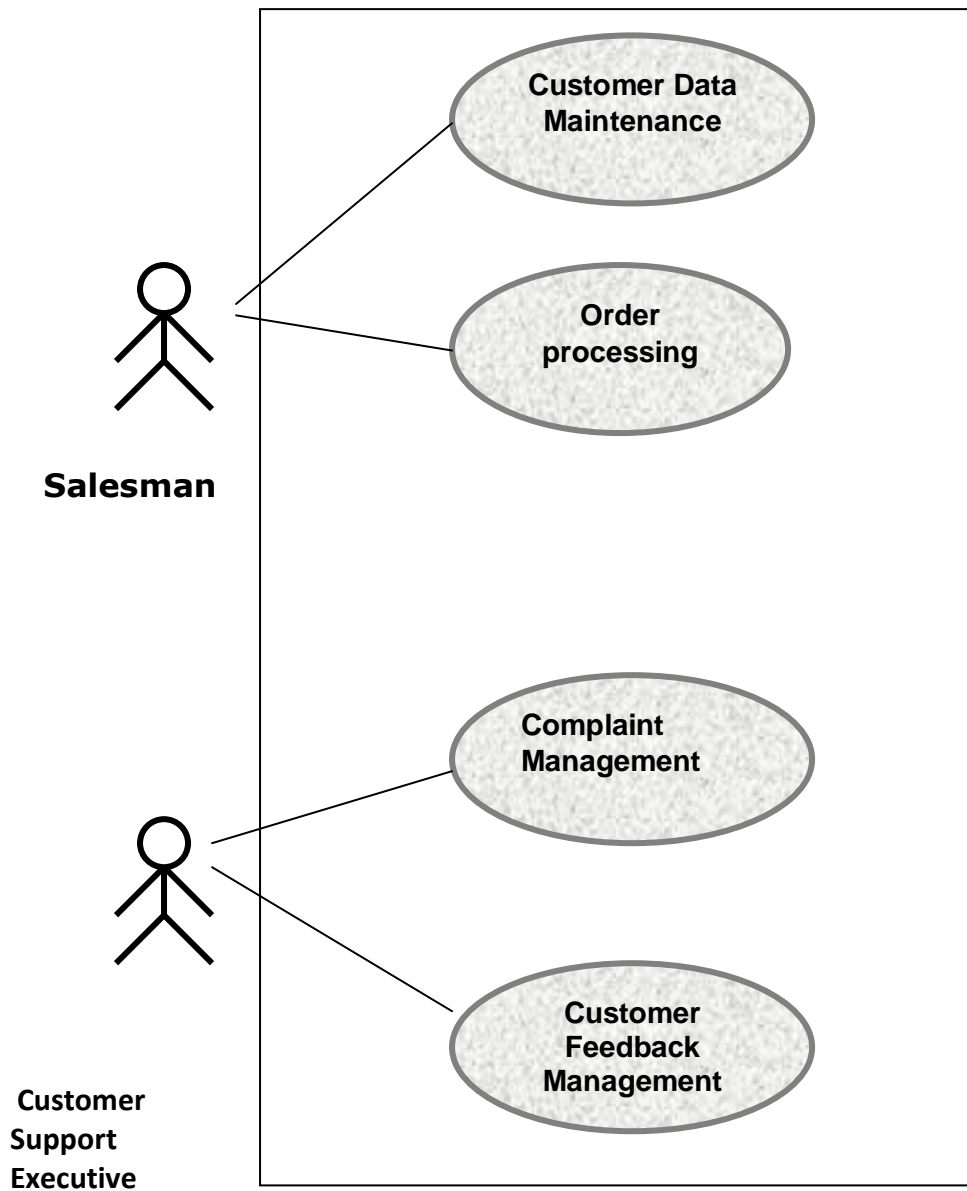


1st Level DFD Of Salesman Data Maintenance Module



ER Diagram of CRM: Sales Force Automation





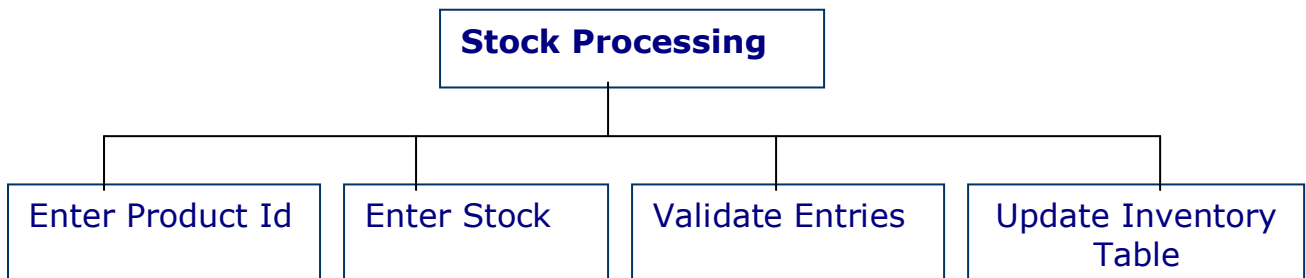
Use Case Diagram of Customer Relationship Management and sales force Management System

HIPO Diagram of Stock Processing Module

IPO-

Input	Process	Output
Product Id Stock	<ul style="list-style-type: none">• Enter Product Id• Enter Stock• Update inventory Table	Stock report

VTOC-

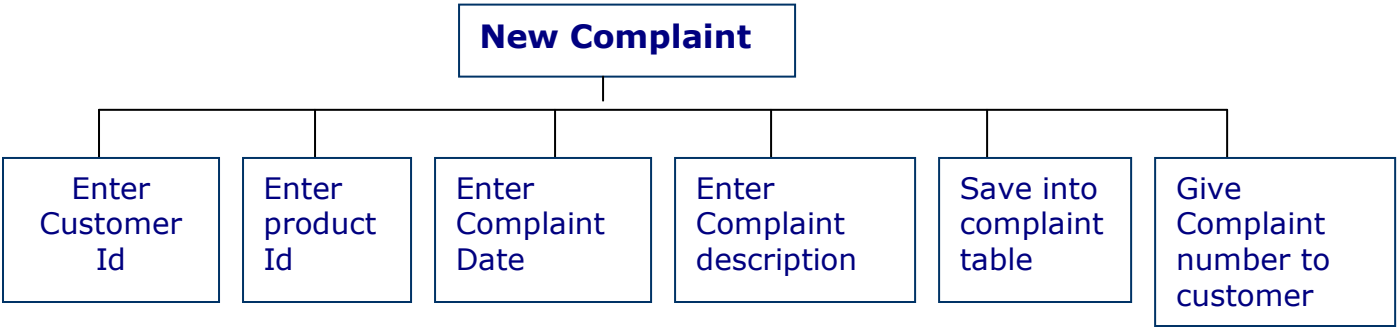


HIPO Diagram of New Complaint Module

IPO-

Input	Process	Output
Customer Id Product Id Complaint Date Complaint description	<ul style="list-style-type: none">Enter Customer IdEnter Product IdEnter Complaint DescriptionEnter date of complaint	Complaint Id

VTOC

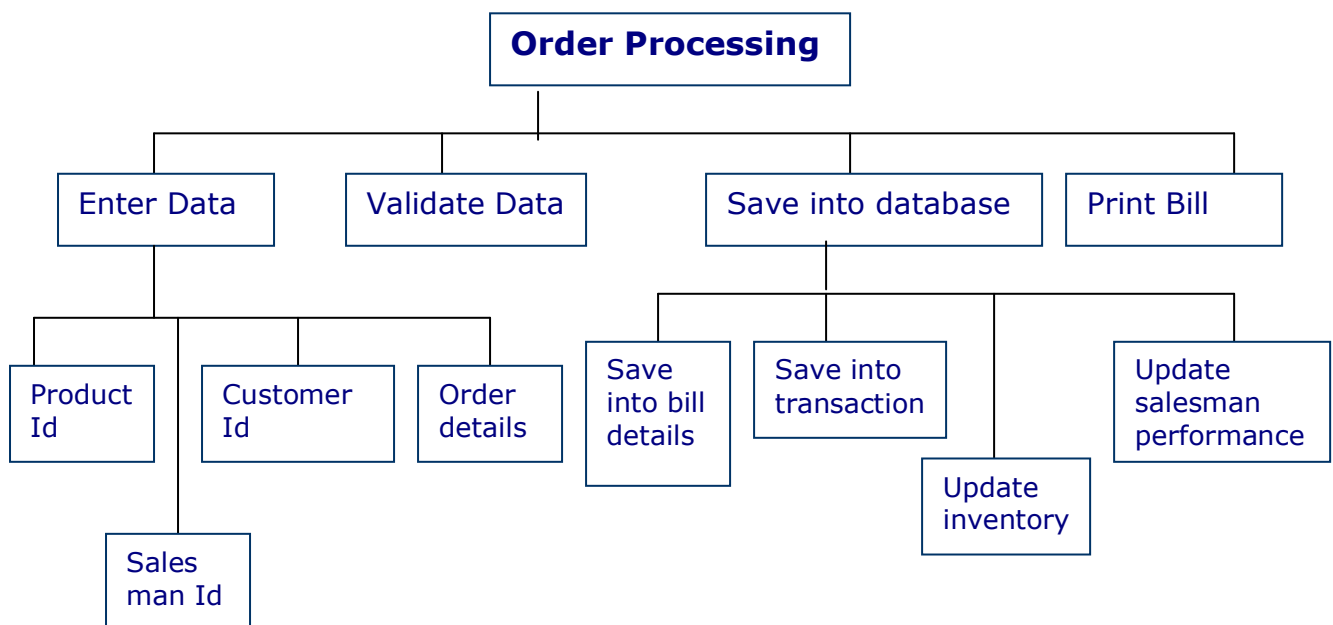


HIPO diagram of Order Processing Module

IPO-

Input	Process	Output
Customer Id Product Id Salesman Id Bill Date Quantity Price of Product	<ul style="list-style-type: none"> • Enter Customer Id • Enter Salesman id • Enter Product Id • Enter Quantity • Enter Date of order • Check product in Inventory • Save into Bill details table • Save into transaction table • Update inventory • Update salesman performance table 	Bill

VTOC-

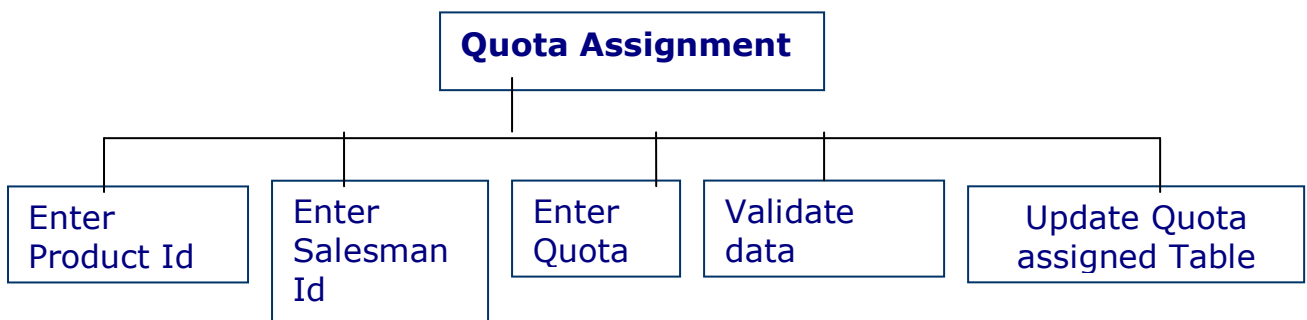


HIPO Diagram of Quota Assignment

IPO

Input	Process	Output
Salesman id Product Id Quota	<ul style="list-style-type: none">• Enter Salesman Id• Enter Product id• Enter Quota• Update Quota Assigned Table	Quota Report

VTOC

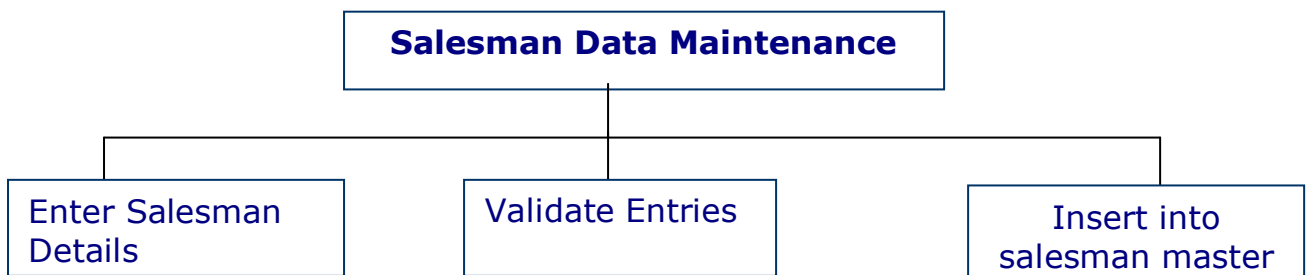


HIPO Diagram Of Salesman Maintenance Module

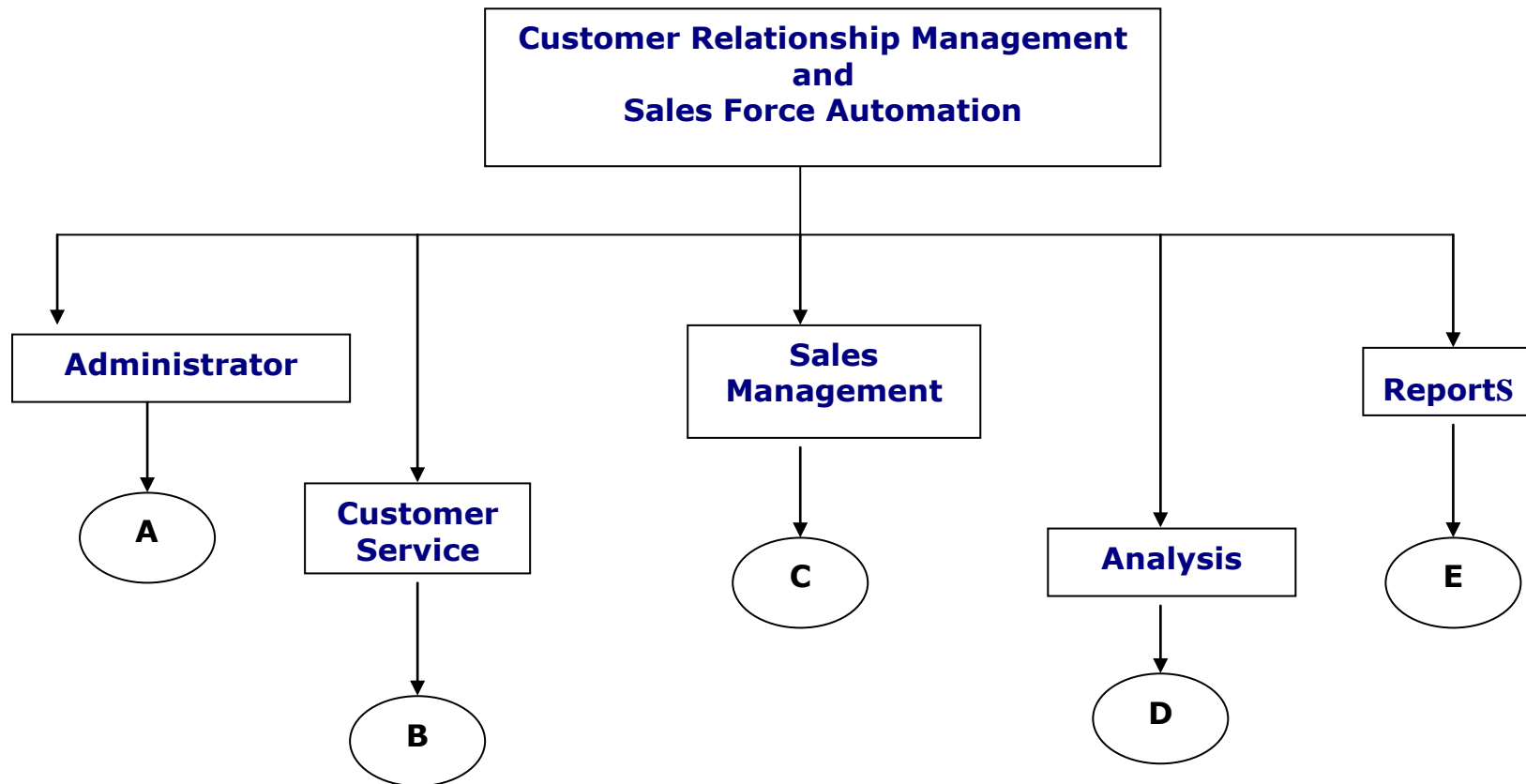
IPO-

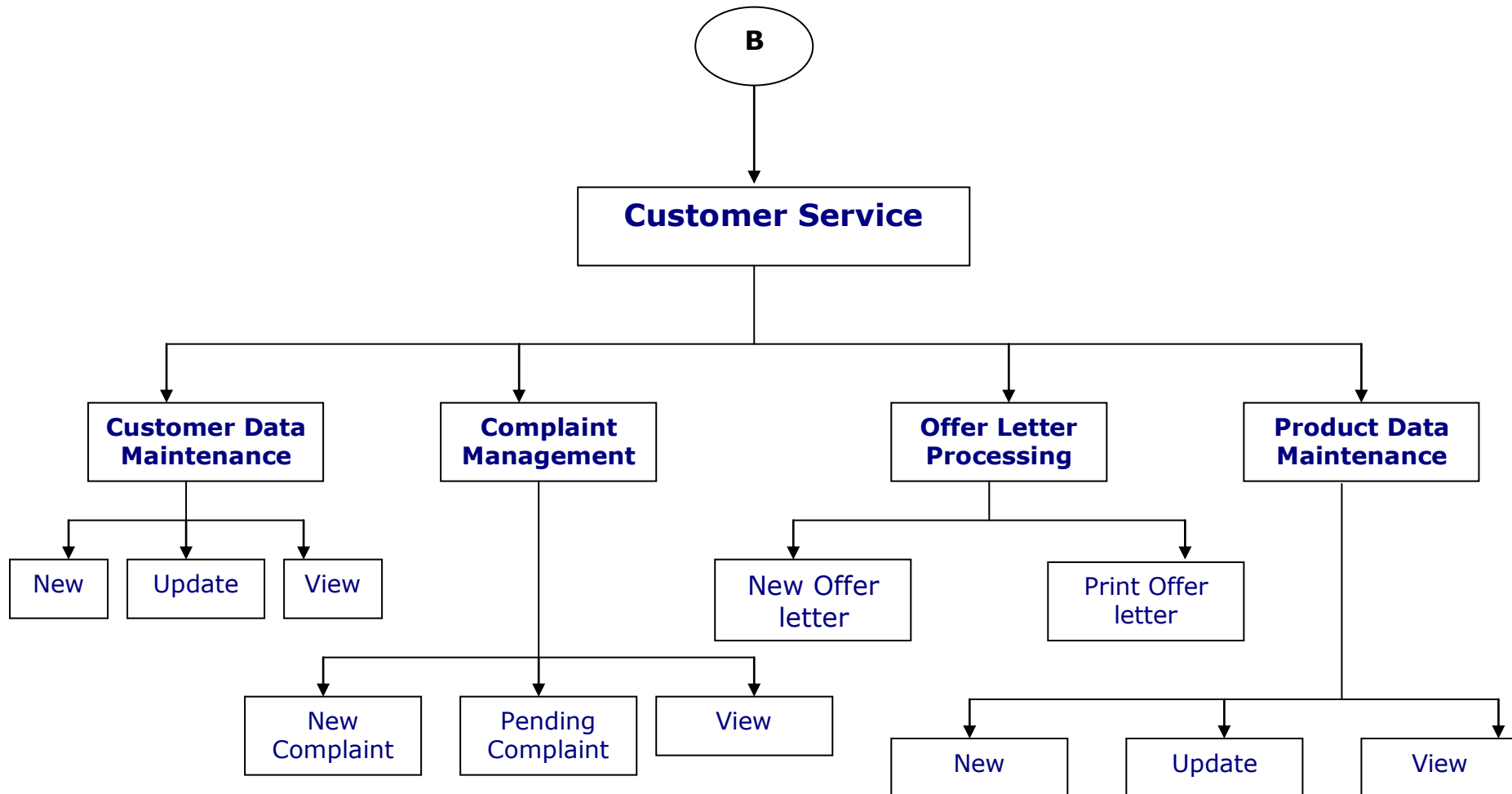
Input	Process	Output
Salesman id Name Address City Phone No. Mobile Email Territory Date of Birth Date of join	<ul style="list-style-type: none"> • Enter Salesman Id • Enter Name • Enter address • Enter City • Enter phone no. • Enter mobile no • Enter Email Id • Enter Territory • Enter Date of birth • Enter Date of Join • Validate Data • Save into salesman master table 	Salesman report

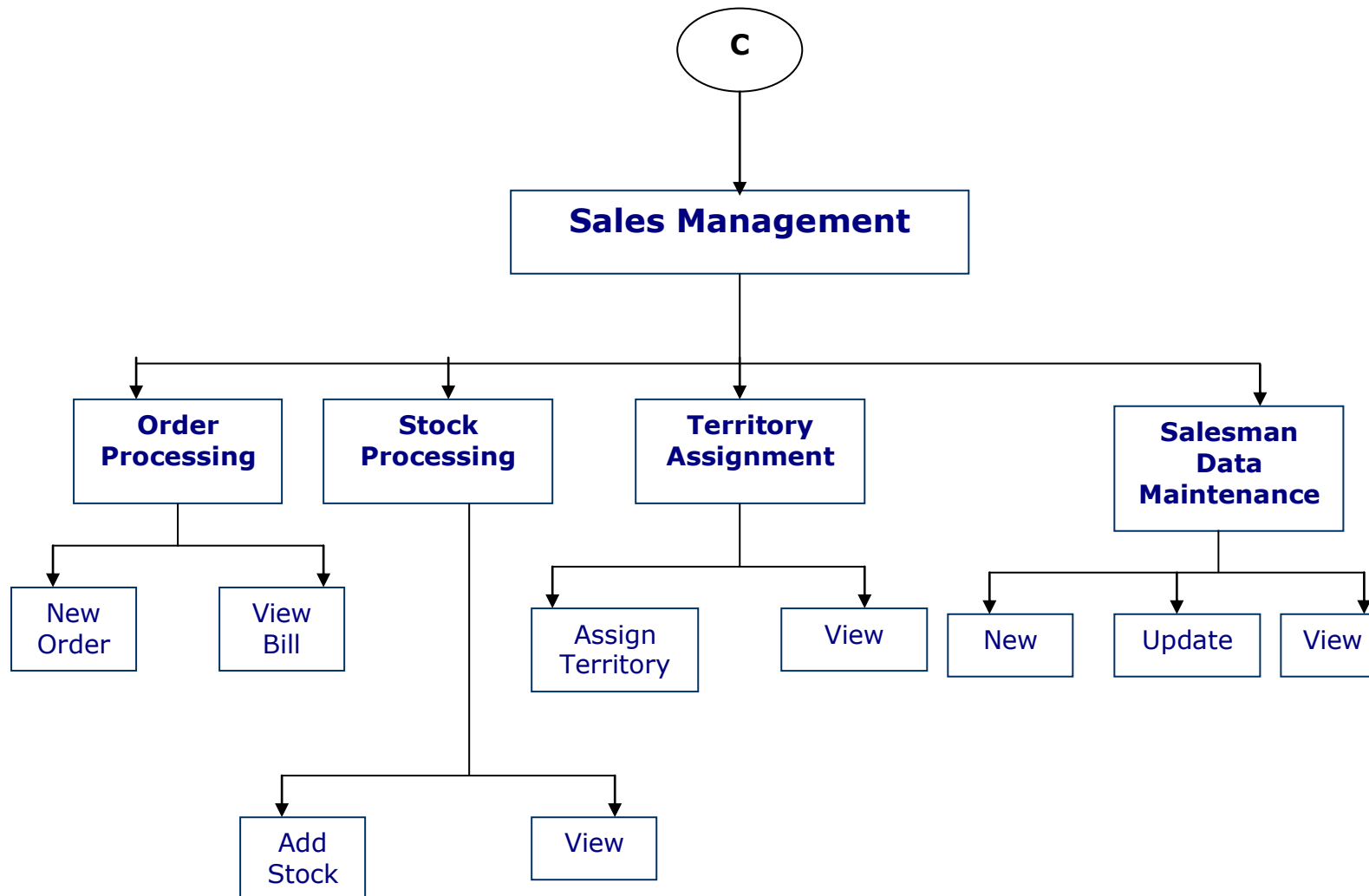
VTOC-

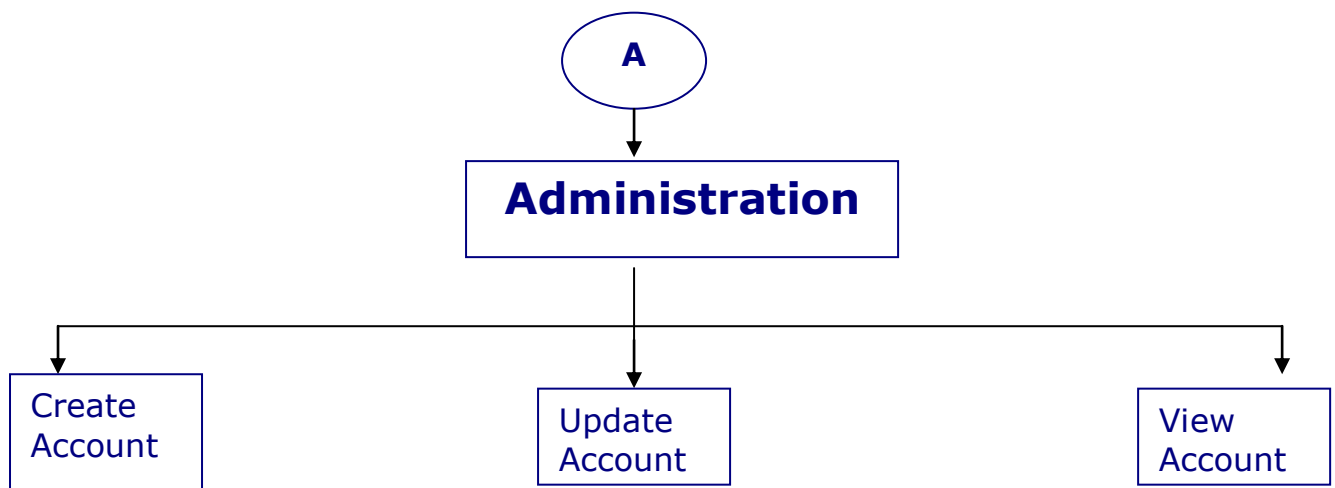
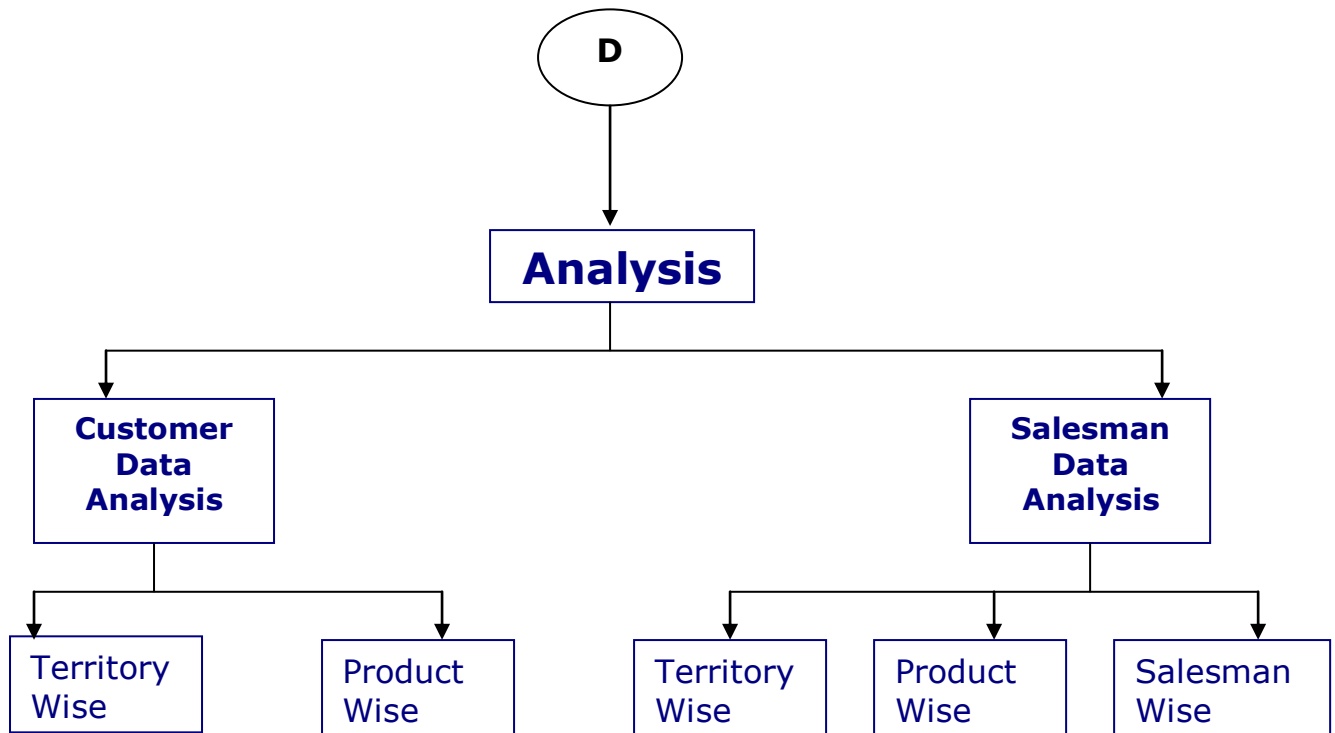


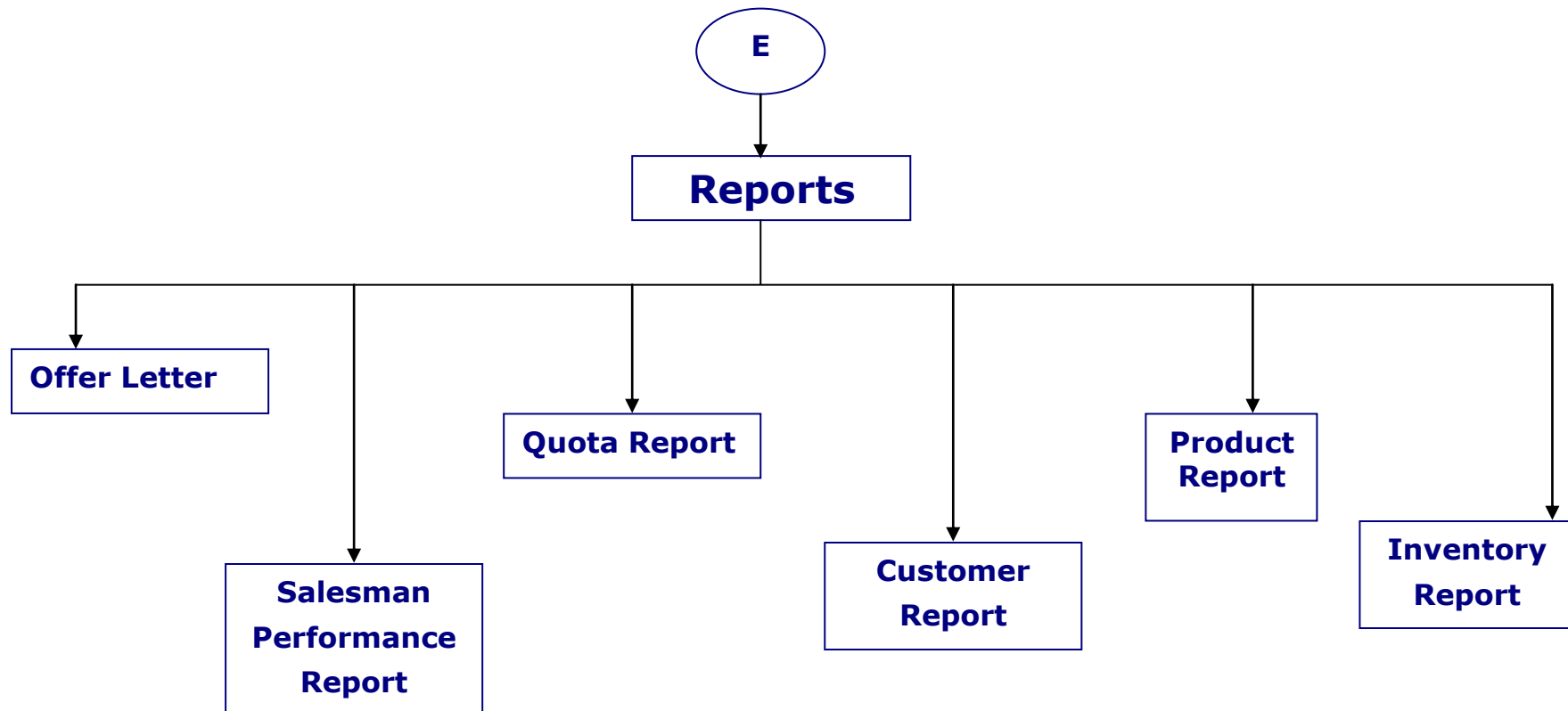
System Chart











CODING

Coding is the phase of Software Life Cycle that produces the Actual code that will be delivered to the customer as the running System. The other phases of the life cycle may also develop code, such as prototypes, tests, and test drivers, but these are for the use by the developer. Individual modules developed in this system are also tested before being delivered to the next phase.

The design must be translated into a machine- readable form. This is what coding. Coding is basically translating the design into a machine-readable form. The code generation step performs this task. If design is performed in a detailed manner, code generation can be accomplished mechanistically.

While converting design into coding, following points are to be considered:

- ☐ Is a Code a proper translation of design?
- ☐ Have the Coding standards been followed?
- ☐ Are the data types and data declarations proper?
- ☐ Are the physical constants correct?

CODE EFFICIENCY

The next thing, which we have to consider, is the “Code Efficiency”. The code that we write should be well efficient and error free. Because as much our code is efficient and error free, our software will be that much effective.

The Efficiency of the code or program depends on:

Readability:

Coding should be much and more readable as in 4GL such as Coding in Visual C++,Java, Oracle,SQL, etc.

Portability:

Coding or program should be written in such a form such that it could be run on different machines with little or no change. Thus, our coding should be as much as portable as it can.

Easy Debugging:

Coding should be such that errors could easily be removed or debugged. That is our coding should be as much as error free as possible.

Easy Software Development:

Software could easily be developed. Commands of programming language are similar to natural languages (English).

Short & Precise Coding:

Coding should not be lengthy. It should be short & precise. This factor is most important for the efficient coding.

OPTIMIZATION OF CODE

Code Optimization is the last & final step in Code generation phase. Code optimization comes after the code generation. We have to optimize the code after generating it.

Code optimization also plays a vital role as the efficiency of code plays in the successful development of the system. Code Optimization also effects the efficiency of code. Our code will be more efficient if it is more optimized.

To produce faster and more compact, the code generator should include some form of “Code Optimization”. This may exploit techniques such as the use of special purpose machine instructions or addressing modes, register optimization, etc. this code optimization may incorporate both machine dependent and machine independent techniques.

VALIDATION

Validation refers to the process of using software in a live environment to find errors.

Validation refers to the set of activities that ensure that the software that has been built is traceable to customer requirements. In validation testing, requirements are established as part of software requirement analysis are validated against the software that has been constructed. The feedback from the validation phase generally brings some changes in the software to deal with errors and failures that are uncovered. Then a set of user sites is selected for putting the

system into use on a live basis. These beta test sites use the system in day-to-day activities; they process live transactions and produce normal system output. Validation Checks may continue for several months depending on the requirements of the end-users for whom the Software is developed. During the course of validating the system, failure may occur and the software will be changed. Continue use may bring more failures and the need for still more changes.

In the last stage of software development, software is completely assembled as a package, some interfacing errors have been uncovered and corrected, and a final series of software tests – validation testing or validation checks may begin. Validation succeeds when the software functions in a manner that can be reasonably expected by the customer.

Validation Criteria is the most important and ironically, the most often neglected section of Software requirements Specification.

Software validations are achieved through a series of black-box tests that demonstrate conformity with requirements. A test plan outlines the classes of tests to be conducted and a test procedure define test cases that will be used to demonstrate conformity with requirements. Both the plan and procedure are designed to ensure that all performance requirements are attained, documentation is correct, and human-engineered and other requirements are met (e.g., transportability, compatibility, error recovery, maintainability).

In computing, the process of checking input data to ensure that it is complete, accurate, and reasonable. Although it would be impossible to guarantee that only valid data are entered into a computer, a suitable combination of validation checks should ensure that most errors are detected.

Validation checks in the project-

There is a package in the project, `validate_data` which contains various functions for validation on different field. Some function used for validation check in the system are-

- **boolean isNotAlphabets(String)**- this function checks that given string contain alphabets only or not. For example in case of Salesman name field, it should contain alphabets only not numeric or special character.
- **boolean isNumber(String)**- this function return true if passed string contain number only otherwise false. In case of phone no, mobile number etc validation is done with this method.
- **boolean isEmail(String)**- this function checks validity of email entered in email field.
- **boolean isDate(String)**- this function checks validity of date entered. If date greater than current date or format of date is not correct or date supplied does not exist than it returns false otherwise true.

Other validation criteria are-

- In the fields where null values are not allowed can not left blank.
- Length of data entered in the system should not have length more than the length specified in the database.

SYSTEM TESTING

Software testing is the process used to help identify the correctness, completeness, security, and quality of developed computer software. Testing is a process of executing a program or application with the intent of finding errors. With that in mind, testing can never completely establish the correctness of arbitrary computer software. In other words, testing is criticism or comparison that is comparing the actual value with an expected one. An important point is that software testing should be distinguished from the separate discipline of software quality assurance, which encompasses all business process areas, not just testing.

TESTING OBJECTIVES :

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has a high-probability of finding an as-yet-undiscovered error.
- A successful test is one that uncovers an as-yet-undiscovered error.

Software Testing Techniques

The importance of software testing and its impact on software cannot be underestimated. Software testing is a fundamental component of software quality assurance and represents a review of specification, design and coding. The greater visibility of software systems and the cost associated with software failure are motivating factors for planning, through testing. It is not uncommon for a software organization to spent 40% of its effort on testing.

White Box Testing

White box testing is a test case design approach that employs the control architecture of the procedural design to produce test cases. Using white box testing approaches, the software engineering can produce test cases that

- (1) Guarantee that all independent paths in a module have been exercised at least once
- (2) Exercise all logical decisions

- (3) Execute all loops at their boundaries and in their operational bounds
- (4) Exercise internal data structures to maintain their validity.

Various white box techniques

Basis Path Testing

Basic path testing is a white box testing techniques that allows the test case designer to produce a logical complexity measure of procedural design and use this measure as an approach for outlining a basic set of execution paths. Test cases produced to exercise each statement in the program at least one time during testing.

Control Structure Testing

Although basis path testing is simple and highly effective, it is not enough in itself. Next we consider variations on control structure testing that broaden testing coverage and improve the quality of white box testing. Different control structure techniques are

- a.* Condition testing
- b.* Data flow testing
- c.* Loop testing

Black Box Testing

Black Box Testing is not a type of testing; it instead is a testing strategy, which does not need any knowledge of internal design or code etc. As the name "black box" suggests, no knowledge of internal logic or code structure is required. The types of testing under this strategy are totally based/focused on the testing for requirements and functionality of the work product/software application.

Various black box techniques

Functional Testing:

In this type of testing, the software is tested for the functional requirements. The tests are written in order to check if the application behaves as expected.

Smoke Testing:

This type of testing is also called sanity testing and is done in order to check if the application is ready for further major testing and is working properly without failing up to least expected level.

Recovery Testing:

Recovery testing is basically done in order to check how fast and better the application can recover against any type of crash or hardware failure etc. Type or extent of recovery is specified in the requirement specifications.

Alpha Testing:

In this type of testing, the users are invited at the development center where they use the application and the developers note every particular input or action carried out by the user. Any type of abnormal behavior of the system is noted and rectified by the developers.

Beta Testing:

In this type of testing, the software is distributed as a beta version to the users and users test the application at their sites. As the users explore the software, in case if any exception/defect occurs that is reported to the developers.

Software Testing Strategies in used in the project

A strategy for software testing integrates software test case design techniques into a well-planned set of steps that cause the production of software. A software test strategy provides a road map for the software developer, the quality assurance organization, and the customer

Unit testing

Unit testing concentrates verification on the smallest element of the program – the module. Using the detailed design description important control paths are tested to establish errors within the bounds of the module.

Firstly the unit testing on various modules and sub modules is performed in the project. Different modules are tested with different correct and incorrect data. For example in the order processing module order of 0 product is not allowed so in this case different methods are used to find out whether the modules is performing all processes correctly. All modules are tested to find out that whether they are working properly.

Integration testing :-

Once all the individual units have been tested there is a need to test how they were put together to ensure no data is lost across interface, one module does not have an adverse impact on another and a function is not performed correctly. Integration testing is a systematic approach that produces the program structure while at the same time producing tests to identify errors associated with interfacing.

In this project bottom up integration testing is used. Firstly lower level modules are tested. As modules are integrated bottom up, processing required for modules subordinates to a given level is always available and the need for stubs is eliminated.

Validation testing :-

As a culmination of testing, software is completely assembled as a package, interfacing errors have been identified and corrected, and a final set of software tests validation testing are started. Validation can be defined in various ways, but a basic one is valid succeeds when the software functions in a fashion that can reasonably expected by the customer.

In the first phase of alpha testing, developers test the software using white box techniques. Additional inspection is then performed using black box techniques. This is usually done by a dedicated testing team. This is often known as the second stage of alpha testing.

Unit and integrated tests concentrate on functional verification of a component and incorporation of components into a program structure. Validation testing demonstrates tractability to software requirements, and system testing validates software once it has been incorporated into a larger system.

Test Cases Specification for System Testing

The different conditions that need to be tested, along with the test cases used for testing those conditions and expected output are given. The goal is to test different functional requirements, as specified in requirement document. Test cases have been selected for both valid and invalid inputs.

Test Cases for Adding New Customer Data:

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	TCustName	Cust_name=""	Customer name is blank
2.	TAddress	Address=""	Address is blank
3.	CCity	City's selected index is 0	Select city
4.	TPhone	Check whether given phone contain only numbers or not	Phone number should contain number only
5.	TMobile	Check whether given phone contain only numbers or not	Mobile number should contain number only
6.	Tmobile	Length is greater than 11	Length of number should be 10
7.	CSalary	Selected index is 0	Select salary
8.	COccupation	Selected index is 0	Select Occupation
9.	TDOB	Check whether date is in correct format	Input Date in correct format

Test Cases for Order Processing Module

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	TBillDate	Format of Date is correct or not	Input date in correct format
2.	TbillDate	Date is blank	Enter Bill Date
3.	CCustId	Customer Id's selected index is 0	Select Customer Id
4.	CSId	Salesman id's selected index is 0	Select salesman Id
5.	CProdId	Product Id's selected index is 0	Select product id

6.	TQuantity	Quantity is blank or 0	Enter Quantity correctly
7.	Count	Count=0	Enter product details

Test Cases for Stock Processing Module

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	CProdId	Product id's selected index is 0	Select product id
2.	TStock	Stock is 0	Enter stock
3.	TStock	Stock field contain alphabet or special char	Enter stock correctly

Test Cases for Quota Assignment Module:

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	CProdId	Product id's selected index is 0	Select product id
2.	CSId	Salesman id's selected index is 0	Select salesman Id
3.	TQuota	Quota is blank	Enter Quota
4.	TQuota	Quota Contain alphabets	Enter Quota Correctly

Test Cases for New Complaint Module

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	CProdId	Product id's selected index is 0	Select product id
2.	TCustId	Customer Id's selected index is 0	Select customer Id
3.	TDescription	Description is blank	Enter Complaint Descriptions
4.	TDOC	Date of complaint is blank	Enter complaint date
5.	TDOC	Date of complaint is in wrong format	Enter date in correct format

Test Cases for Pending Complaint module

Seq. No	Test Fields	Condition Being Checked	Expected Output
1.	CCompId	Complaint Id is not selected	Select Complaint id
2.	TDateAction	Date is in wrong format	Enter Date of action correctly
3.	TTectnician	Technician name is blank	Enter technician name
4.	TActionTaken	Action taken is blank	Enter action taken to solve the complaint

SYSTEM IMPLEMENTATION

The purpose of **System Implementation** can be summarized as follows: making the new system available to a prepared set of users (the deployment), and positioning on-going support and maintenance of the system within the Performing Organization (the transition). At a finer level of detail, deploying the system consists of executing all steps necessary to educate the Consumers on the use of the new system, placing the newly developed system into production, confirming that all data required at the start of operations is available and accurate, and validating that business functions that interact with the system are functioning properly. Transitioning the system support responsibilities involves changing from a system development to a system support and maintenance mode of operation, with ownership of the new system moving from the Project Team to the Performing Organization.

Once the design model of **“CRM: Sales Force Automation System”** is created, it is implemented as a prototype, examined by end-users and modified by developers, i.e. us, based on their comments. To accommodate this iterative design approach, a broad class of interface design and prototyping tools has evolved.

This phase consists of the following processes:

- **Prepare for System Implementation**, where all steps needed in advance of actually deploying the application are performed, including preparation of both the production environment and the Consumer communities.
- **Deploy System**, where the full deployment plan, initially developed during System Design and evolved throughout subsequent lifecycle phases, is executed and validated. _
- **Transition to Performing Organization**, where responsibility for and ownership of the application are transitioned from the Project Team to the unit in the Performing Organization that will provide system support and maintenance.

Prepare for system implementation

In the implementation of any new system, it is necessary to ensure that the Consumer community is best positioned to utilize the system once deployment efforts have been validated. Therefore, all necessary training activities must be scheduled and coordinated. As this training is often the first exposure to the system for many individuals, it should be conducted as professionally and competently as possible. A positive training experience is a great first step towards Customer acceptance of the system. Often the performance of deployment efforts impacts many of the Performing Organization's normal business operations. Examples of these impacts include:

- Consumers may experience a period of time in which the systems that they depend on to perform their jobs are temporarily unavailable to them.
- Technical Services personnel may be required to assume significant implementation responsibilities while at the same time having to continue current levels of service on other critical business systems.
- Technical Support personnel may experience unusually high volumes of support requests due to the possible disruption of day-to-day processing.

DEPLOY SYSTEM

Deploying the system is the culmination of all prior efforts where all of the meetings, planning sessions, deliverable reviews, prototypes, development, and testing pay off in the delivery of the final system. It is also the point in the project that often requires the most coordination, due to the breadth and variety of activities that must be performed. Depending upon the complexity of the system being implemented, it may impact technical, operational, and cultural aspects of the organization. A representative sample of high-level activities might include the installation of new hardware, increased network capabilities, deployment and configuration of the new system software, a training and awareness campaign, activation of new job titles and responsibilities, and a completely new operational support structure aimed at providing Consumer-oriented assistance during the hours that the new system is available for use (to name a few).

TRANSITION TO PERFORMING ORGANIZATION

In many organizations, the team of individuals responsible for the long-term support and maintenance of a system is different from the team initially responsible for designing and developing the application. Often, the two teams include a comparable set of technical skills. The responsibilities associated with supporting an operational system, however, are different from those associated with new development. In order to affect this shift of responsibilities, the Project Team must provide those responsible for system support in the Performing Organization with

a combination of technical documentation, training, and hands-on assistance to enable them to provide an acceptable level of operational support to the Consumers. This system transition is one element

SYSTEM MAINTENANCE

Software once designed & developed needs to be maintained till end of software / system life. When the system is fully implemented, analyst must take precautions to ensure that the need for maintenance is controlled through design and testing and the ability to perform it is provided through proper design practices.

Maintenance is necessary to eliminate errors in the working system during its working life and to tune the system to any variations in its working environment. Often small system deficiencies are found as a system is brought into operations and changes are made to remove them. System planners must always plan for resource availability to carry out these maintenance functions. The importance of maintenance is to bring the new system to standards.

The term **“Software Maintenance”** is commonly used to refer to the modification that are made to a software system in its initial release.

Maintenance requirements for information systems

- (a) From 60% to 90% of the overall cost of software during the life of a system is spend on maintenance.
- (b) Often maintenance is not done very efficiently.
- (c) Software demand is growing at a faster rate than supply. Many programmers spending more time on systems maintenance than on new software development.

The keys to reduce the need of maintenance , while making it possible to do essential tasks more efficiently, are as follows :

- More accurately defining the user's requirements during systems development.
- Making better Systems documentation.
- Using proper methods of designing processing logic and communicating it to project team members.
- Utilising the existing tools and techniques in an effective way.
- Managing the systems engineering process in a better and effective way.

SYSTEM SECURITY MEASURES

The System designed & developed is to be kept secure from various persons or users who are not allowed to use the software. Various Security measures are taken to prevent the Software / System from other persons or users.

In this System, users are divided in three category-

- **Administrator** has authority of maintaining salesman data, product data, assigning territory to the salesman, stock processing, offer letter processing, quota assignment, creating user account. Administrator can also provide facility of analyzing customer, sales, and complaint data.
- **Salesman** has authority of adding customer details, order processing, sales call data management.
- **Customer Support executive** has authority of complaint processing and entering customer feedback.

Different authority is assigned different user. Each user has a user name and password to access the system. Every time when user will open the system, user has to give his/her username and password in the login window. Access category is assigned to every user. On the basis of this category user get access in the system.

COST ESTIMATION OF THE PROJECT

The most fundamental calculation in the COCOMO model is the use of the Effort Equation to estimate the number of Person-Months required to develop a project. Most of the other COCOMO results, including the estimates for Requirements and Maintenance, are derived from this quantity.

The COCOMO calculations are based on your estimates of a project's size in Source Lines of Code (SLOC). SLOC is defined such that:

- Only Source lines that are DELIVERED as part of the product are included -- test drivers and other support software is excluded
- SOURCE lines are created by the project staff -- code created by applications generators is excluded
- One SLOC is one logical line of code
- Declarations are counted as SLOC
- Comments are not counted as SLOC

In the COCOMO II model, some of the most important factors contributing to a project's duration and cost are the Scale Drivers. You set each Scale Driver to describe your project; these Scale Drivers determine the exponent used in the Effort Equation.

The 5 Scale Drivers are:

- Precedentedness
- Development Flexibility
- Architecture / Risk Resolution
- Team Cohesion
- Process Maturity

In COCOMO II Model Effort is calculated as following-

$$\text{Effort} = 2.94 * \text{EAF} * (\text{KSLOC})^E$$

Where

EAF Is the Effort Adjustment Factor derived from the Cost Drivers

E Is an exponent derived from the five Scale Drivers

In the COCOMO Model phase wise distribution of effort is divided in different category.

<i>Effort Category</i>	<i>KDLOC</i>
<i>Small</i>	<i>2</i>
<i>Intermediate</i>	<i>8</i>
<i>Medium</i>	<i>32</i>
<i>Large</i>	<i>128</i>

In this project it is estimated that line of codes will be around 10 thousand lines. Now on the basis of 8 KDLOC E (Effort) has been calculated.

Before the calculation of E it is required to calculate EAF (Effort Adjustment Factor).

To calculate the EAF we have to collect the rating of cost Drivers as follows:

Complexity	High	1.15
Storage	High	1.06
Experience	High	0.91
Programmer Capability	High	0.86
Database Size	Nominal	1.00

$$\text{EAF} = 1.5 * 1.06 * .91 * .86 * 1 = 1.24$$

$$\text{Effort} = 2.94 * (1.24) * (10)^{1.0997} = 45.86 \text{ Person-Months}$$

$$\text{Duration} = 3.67 * (\text{Effort})^{SE}$$

Here SE=.3179

$$\text{Duration} = 3.67 * 45.86^{.3179} = 12.38 \text{ Months}$$

So, in this manner we have calculated the total duration of the Project. And the Total Duration Comes Near **About 12.38 Month** to complete the Project.

It is analyzed that the project will benefit the organization in terms of time saving as well as cost wise. It is going to improve accuracy at each phase of the system. If we consider the Benefits gained by the system we can see that the software is feasible in economical point of view.

REPORTS

Various Reports are generated in this project-

1. Salesman Performance Report
2. Customer Report
3. Product Report
4. Inventory report
5. Complaint Reports
6. Quota Report
7. Offer Letter
8. Bill
9. Analysis Reports

1. Salesman Performance Report-

This report will display performance of the sales person. It will check display both standard performance and actual performance of sales person. With the help of this report administrator can get information that what quota assigned to salesman and what is the actual sale done by the salesman.

2. Customer Report

This report generate list of customer on the basis of demand submitted by the user. For example if user want list of particular city than he can submit it and he will obtain it. Report generation on the basis of multiple criteria is also possible such as on the basis of city, income group and Occupation.

3. Product Report

This report will generate detail product description and list of product list on the demand of customer.

4. Inventory Report

This report provides information related to inventory of different products.

5. Quota Report-

This report will display details of how much quota assigned to which sales person. Administrator has to provide salesman Id the report will generate accordingly.

6. Complaint Report

This report will generate detail of complains and their solution for a particular period. This report will show details of different types of complain, which product has maximum number of complains and which product has no complain for particular period.

7. Offer Letter

Offer Letter is generated on the basis of condition provided. This report will display details of offer letter along with details of customer to whom it will be send.

8. Bill

Bill is generated on the time of sale of any product. Bill contains information related to customer, salesman, and product details.

9. Analysis Reports

Administrator can get various analysis report on the basis of sales data, customer data.

PERT CHART

A PERT chart is a project management tool used to schedule, organize, and coordinate tasks within a project. PERT stands for Program Evaluation Review Technique. A similar methodology, the Critical Path Method (CPM) was developed for project management in the private sector at about the same time.

A PERT chart presents a graphic illustration of a project as a network diagram consisting of numbered nodes representing events, or milestones in the project linked by labeled vectors (directional lines) representing tasks in the project.

Steps in the PERT Planning Process

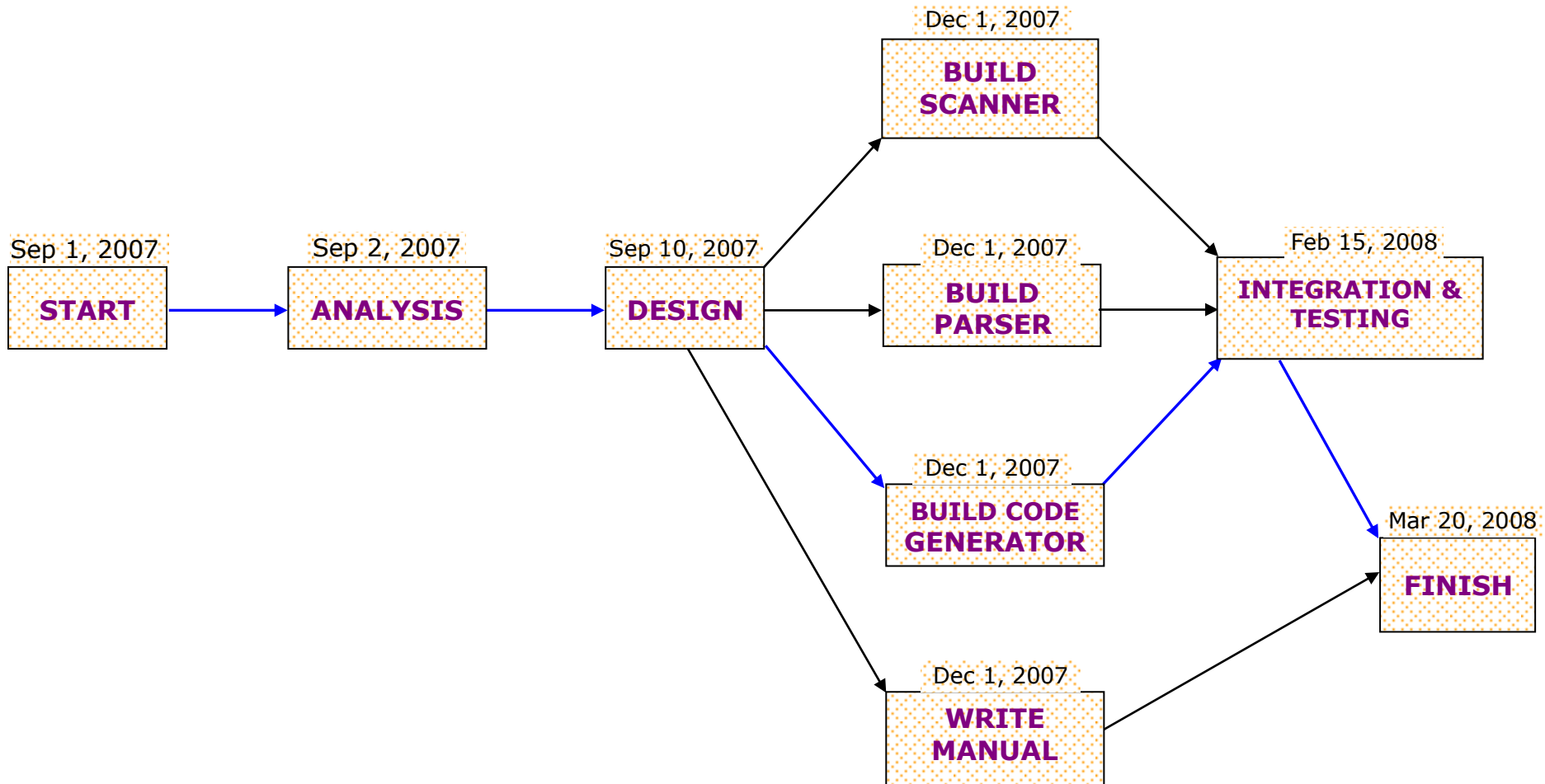
PERT planning involves the following steps:

1. Identify the specific activities and milestones.
2. Determine the proper sequence of the activities.
3. Construct a network diagram.
4. Estimate the time required for each activity.
5. Determine the *critical path*.
6. Update the PERT chart as the project progresses.

PERT is useful because it provides the following information:

- Expected project completion time.
- Probability of completion before a specified date.
- The critical path activities that directly impact the completion time.
- The activities that have slack time and that can lend resources to critical path activities.

PERT CHART FOR “CRM: Sales Force Automation”



The PERT Chart shows that the path through the project that consists of the *design, build code generator* and *integration & testing activities* is the *Critical path* for the project. (shown by blue Arrows in above Chart).

GANTT CHART

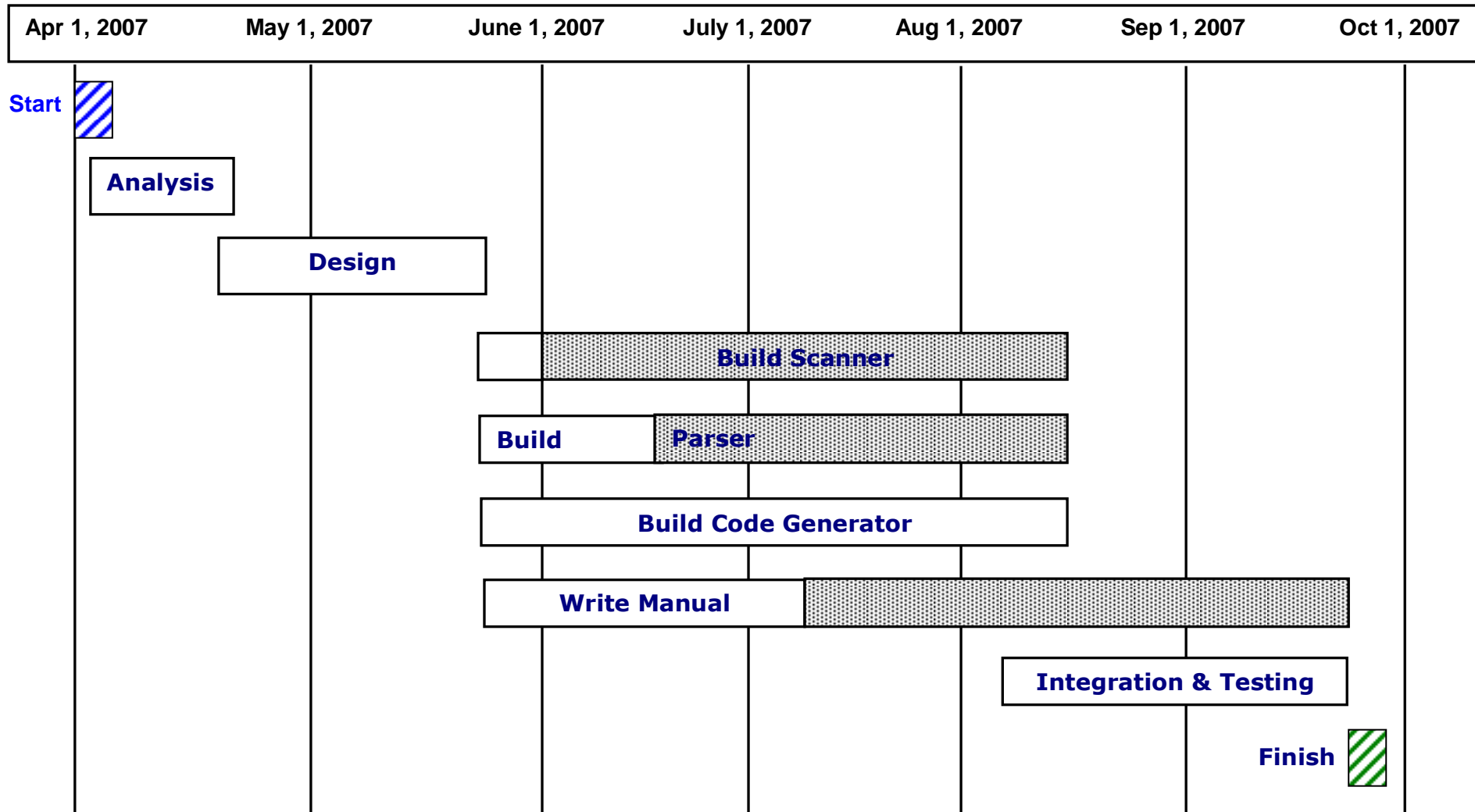
The Gantt chart, developed by Charles Gantt in 1917, focuses on the sequence of tasks necessary for completion of the project at hand. Each task on a Gantt chart is represented as a single horizontal bar on an X-Y chart. The horizontal axis (X-axis) is the time scale over which the project will endure. Therefore, the length of each task bar corresponds to the duration of the task, or the time necessary for completion. Arrows connecting independent tasks reflect the relationships between the tasks it connects. The relationship usually shows dependency where one task cannot begin until another is completed. The resources necessary for completion are also identified next to the chart. The Gantt chart is an excellent tool for quickly assessing the status of a project.

GANTT Charts are a project control technique that can be used for several purposes, including :

- ☐ scheduling,
- ☐ budgeting, and
- ☐ resource planning

A GANTT Chart is a bar chart, with each bar representing an activity. The bars are drawn against a time line. The length of each bar is proportional to the length of time planned for the activity.

GANTT CHART FOR "CRM: Sales Force Automation System "



FUTURE SCOPE OF THE PROJECT

This application can be implemented in those organizations where there is a need for automation of sales work. This application minimizes manual work in the organization so the work load on the organization will decrease and efficiency and effectiveness of the organization will increase. This application helps in reducing redundancy of the data and provides security so that unauthorized persons cannot access the application. The database management in the organization becomes more reliable. This application provides a facility to analyze data of customer, sale and complaints. This can be highly useful in planning marketing and sales strategy of the organization. It attempts to integrate and automate the various customer-serving processes within a company. CRM is a strategy used to learn more about customers' needs and behaviors in order to develop stronger relationships with them. Good customer relationships are at the heart of business success. So this application will help in building good customer relations with customers.

With the help of this project, the organization will be able to manage customer data in an effective way. Proper management of sales and customer data will provide a facility of obtaining customer details time to time on the basis of which the organization can make marketing and sales plans effectively.

FURTHER ENHANCEMENT OF THE PROJECT

There is always room for improving in any software package, however good efficient it may be. But the important thing is that the system should be flexible enough for future modification/alteration whenever and by whomsoever it may be. Keeping in consideration this important factor, the system is designed in such a way. The software is developed in modules are efficient enough to introduce any change in the software to get more information.

- Similarly, the present system can be implemented on Internet and software can be connected to the various branches of this home appliance company of course with more security constraints added to it.
- This project can be attached to website of the company which may provide information related to products and also provide facility of registering product etc. It may also help in finding new prospective customer.
- The backend can be improved using Oracle; it will provide better database management and securities.
- More modules can be added in the system such as it can provide facility of direct email so that organization can generate offer letter and send it directly to the customer.

REFERENCES

Websites

http://en.wikipedia.org/wiki/Customer_relationship_management

<http://www.zeelabs.com/CRM.htm>

<http://www.crmnewz.com/>

<http://www.microsoft.com/crm>

<http://www.crystaldecisions.com/>

www.crmsolution.com

BOOKS

“Analysis And Design Of Information Systems” By **V. Rajaraman**

“Principles Of Information Systems” By **Ralph M.Stair And George W. Reyland**

“Visual C++ Programming” By **Yashwant Kanetkar**

“MFC C++ Classes” By **Wodtke**

“Mastering C++” By **Venugopal, Rajkumar, Ravishanker**

“MFC Programming (Unleased)” By **SAMS Publication**

“Programming Microsoft Visual C++” By **Kruglinski, Wingo**

“The Complete Reference Oracle 10g” By **TMH**

“Crystal Report Developer Guide” By **Businness.com**