

**PANIMALAR INSTITUTE OF TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SEVENTH SEMESTER
SOFTWARE PROJECT MANAGEMENT**

UNIT I - PROJECT EVALUATION AND PROJECT PLANNING

1.1 Project

A project is well-defined task, which is a collection of several operations done in order to achieve a goal (for example, software development and delivery). A Project can be characterized as:

- Every project may has a unique and distinct goal.
- Project is not routine activity or day-to-day operations.
- Project comes with a start time and end time.
- Project ends when its goal is achieved hence it is a temporary phase in the lifetime of an organization.
- Project needs adequate resources in terms of time, manpower, finance, material and knowledge-bank.

1.2 Software Project

A Software Project is the complete procedure of software development from requirement gathering to testing and maintenance, carried out according to the execution methodologies, in a specified period of time to achieve intended software product.

1.3 Importance of software project management

Software is said to be an intangible product. Software development is a kind of all new stream in world business and there's very little experience in building software products. Most software products are tailor made to fit client's requirements. The most important is that the underlying technology changes and advances so frequently and rapidly that experience of one product may not be applied to the other one. All such business and environmental constraints bring risk in software development hence it is essential to manage software projects efficiently.



The image above shows triple constraints for software projects. It is an essential part of software organization to deliver quality product, keeping the cost within client's budget constrain and deliver the project as per scheduled. There are several factors, both internal and external, which may impact this triple constrain triangle. Any of three factors can severely impact the other two. Therefore, software project management is essential to incorporate user requirements along with budget and time constraints.

1.4 Activities Methodologies

1.4.1 Activities of Project Management

Project management plan begins with a set of activities that are involved in the development process.

- Overview of the project
- Project deliverables
- Managerial processes
- Technical processes
- Work packages
- Schedule of the project
- Budget estimation

1.4.2 Characteristics of Project

Some of the characteristics of project include:

- Planning of process is required;
- Clear objectives have to be specified;
- Project must have a predetermined time span;
- Involves different phases of work;
- Resources used on the project are constrained.;
- Non-routine tasks are involved.

1.4.3 Activities Covered by SPM

A software project is considered as a software application with specific elements associated with each type of project. The lists of activities involved in software project management are:

- Feasibility Study;
- Planning Phase;
- Project Execution.

Feasibility Study

A valid business case implies a prospective project. The necessary information required for the proposed application is gathered. Initial requirement stage is quite complex and difficult. The client is aware of the problems but not sure of how to achieve the solution. Estimation becomes an important factor in the development of the product. Developmental and operational costs have to be estimated along with the benefits of the system. For a complex project, the feasibility study can have sub phases and strategic planning becomes essential in prioritizing the range of potential software developments. Group of projects are termed as a planned programme of development.

Feasibility studies aim to objectively and rationally uncover the strengths and weaknesses of the existing business or proposed venture, opportunities and threats as presented by the environment, the resources required to carry through, and ultimately the prospects for success. In its simplest term, the two criteria to judge feasibility are cost required and value to be attained. As such, a well-designed feasibility study should provide a historical background of the business or project, description of the product or service, accounting statements, details of the operations and management, marketing research and policies, financial data, legal requirements and tax obligations. Generally, feasibility studies precede technical development and project implementation.

Technology and System Feasibility

The assessment is based on an outline design of system requirements in terms of Input, Processes, Output, Fields, Programs, and Procedures. This can be quantified in terms of volumes of data, trends, frequency of updating, etc. in order to estimate whether the new system will perform adequately or not. Technological feasibility is carried out to determine

whether the company has the capability, in terms of software, hardware, personnel and expertise, to handle the completion of the project when writing a feasibility report, the following should be taken to consideration:

- A brief description of the business
- The part of the business being examined
- The human and economic factor
- The possible solutions to the problems

At this level, the concern is whether the proposal is both technically and legally feasible.

Economic Feasibility

Economic analysis is the most frequently used method for evaluating the effectiveness of a new system. More commonly known as cost/benefit analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system. An entrepreneur must accurately weigh the cost versus benefits before taking an action.

Cost-based study: It is important to identify cost and benefit factors, which can be categorized as follows: Development costs and Operating costs. This is an analysis of the costs to be incurred in the system and the benefits derivable out of the system.

Time-based study: This is an analysis of the time required to achieve a return on investments. The future value of a project is also a factor.

Legal Feasibility

Legal feasibility determines whether the proposed system conflicts with legal requirements, e.g. a data processing system must comply with the local Data Protection Acts.

Operational Feasibility

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

Schedule Feasibility

A project will fail if it takes too long to be completed before it is useful. Typically this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is.

Planning Phase

The planning phase comes into existence only if the proposed project is a prospective one. This is found only by the outcome of the feasibility study phase. In case of complex project, a detailed plan is not needed during the initial stage of planning phase. Instead, an outline plan is formulated for the whole project except for the first phase, which has a detailed one. As the project steps into different phases, a detailed plan for each stage can be developed as they are approached this will provide a clear idea about what should be done at every stages of the development.

The **Project Planning Phase** is the second phase in the *project life cycle*. It involves creating of a set of plans to help guide your team through the execution and closure phases of the project. The plans created during this phase will help you to manage time, cost, quality, change, risk and issues. They will also help you manage staff and external suppliers, to ensure that you deliver the project on time and within budget.

In the Planning Phase, the team defines the solution in detail what to build, how to build it, who will build it, and when it will be built. During this phase the team works through the design process to create the solution architecture and design, writes the functional specification, and prepares work plans, cost estimates, and schedules for the various deliverables.

The Planning Phase culminates in the Project Plans Approved Milestone, indicating that the project team, customer, and key project stakeholders agree on the details of the plans. Plans prepared by team members for areas such as communications, test, and security, are rolled up into a master plan that the program manager coordinates. The team's goal during this phase is to document the solution to a degree that the team can produce and deploy the

solution in a timely and cost-effective manner. These documents are considered living documents, meaning they will be updated continuously throughout the Planning Phase.

Diligent work in the Planning Phase, which often involves several iterations of plans and schedules, should mitigate risks and increase chances for success. The team continues to identify all risks throughout the phase, and it addresses new risks as they emerge.

Project Execution

There are two phases of project execution namely design and implementation. The boundary between these two phases must be clearly understandable. Design is about thinking and decision making about the form of the products which has to be created. Implementation lays down the activities that have to be carried out to create these products. Planning and design phase are difficult to separate at the most detailed level because planning decisions are influenced by design decisions. For example, if a software product development has five components then it must have five sets of activities defined for each component.

Project execution is the process from after the contract is signed to the point where the technology is ready for operational use. New and modified products must be ready from a technological and operational point of view before installation and operational use. This is achieved by carrying out the project planning process followed by the project execution process. A successful project execution process will make a new or modified product ready from a technological and operational point of view.

The project planning process will identify technical gaps related to the product itself, environment, standards, governing documents, verification, handling and documentation. The technology qualification program (TQP) is a project plan that describes activities and decision gates for a specific product required to close these gaps.

The project planning process may also identify gaps related to vendor's organization. These gaps must be corrected prior to project execution and is not a part of the TQP. A preliminary TQP will be worked out by the vendor as a part of their tender. The TQP will be finalized in co-operation with the operator prior to contract award. There will be no need for

the TQP when a product can be delivered off the shelf in accordance with operator's technical requirements.

The TQP describes required activities related to 'development and qualification testing (QT) in the above figure. Technology readiness is achieved when the TQP activities are executed and accepted.

The manufacturing and factory acceptance testing (FAT) is controlled by the quality plan. The operational preparations are controlled by the operational manager. Operational readiness is achieved when the manufacturing and operational preparations are finalized and accepted.

Vendors have quality assurance (QA) systems to provide quality in all steps of their services. These QA systems shall be used to establish the TQP and quality plans during the project planning process. Operators have requirements and recommended practices that shall be used during the operational preparation process. Still there is need for a practical summary of the entire project execution process as it will be for new technology. Such summary is wanted by completion- and drilling engineers responsible for the project planning process and will be used to control the content of the TQP and quality plan worked out by the vendors.

This need has resulted in the development of a guideline describing the entire project execution process. The guideline is fitted to operator needs and has thus emphasis on qualification activities. The guideline is made for well technology, but the main principles can be used for most technology elements.

1.4.4 Plan, Methods & Methodologies

An **activity plan** is based on some method of work. To test software the following list is assumed.

- Requirement analysis for the software;
- Develop test cases for each requirement;
- Creating test scripts , expected results ;
- Comparison of actual result with the expected result;
- Identifying the discrepancies.

A **method** denotes a kind of activity.

A **plan** takes the method and converts it to activities. Every activity identified must contain the start and end dates, the responsible person to carry out the activity, what tools and materials are used.

Complex procedures can be handled in sequence or in parallel manner. The output of first method will be the input of the second method, the second one's output might be the input of the third one and so on. Methods grouped together are termed as methodologies. For example, object oriented design is a methodology made up of several methods.

In strategic planning, resource allocation is a plan for using available resources, for example human resources, especially in the near term, to achieve goals for the future. It is the process of allocating resources among the various projects or business units. The plan has two parts: Firstly, there is the basic allocation decision and secondly there are contingency mechanisms. The basic allocation decision is the choice of which items to fund in the plan, and what level of funding it should receive, and which to leave unfunded: the resources are allocated to some items, not to others.

There are two contingency mechanisms. There is a priority ranking of items excluded from the plan, showing which items to fund if more resources should become available; and there is a priority ranking of some items included in the plan, showing which items should be sacrificed if total funding must be reduced.

1.4.5 Software Process/Phases of SPM

1. Project conception and initiation

An idea for a project will be carefully examined to determine whether or not it benefits the organization. During this phase, a decision making team will identify if the project can realistically be completed.

2. Project definition and planning

A project plan, project charter and/or project scope may be put in writing, outlining the work to be performed. During this phase, a team should prioritize the project, calculate a budget and schedule, and determine what resources are needed.

3. Project launch or execution

Resources' tasks are distributed and teams are informed of responsibilities. This is a good time to bring up important project related information.

4. Project performance and control

Project managers will compare project status and progress to the actual plan, as resources perform the scheduled work. During this phase, project managers may need to adjust schedules or do what is necessary to keep the project on track.

5. Project close

After project tasks are completed and the client has approved the outcome, an evaluation is necessary to highlight project success and/or learn from project history. Projects and project management processes vary from industry to industry; however, these are more traditional elements of a project. The overarching goal is typically to offer a product, change a process or to solve a problem in order to benefit the organization.

1.5 Categorization of Software Projects

The Categories are

- Compulsory versus voluntary users
- Information System versus Embedded Systems
- Outsources Projects
- Objective-driven development

Information system interfaces with the organization whereas an embedded system interfaces with a machine. Typical example for an information system can be inventory system maintained by an organization. An embedded system or an industrial system can be a process control system such as maintaining air conditioning equipment in a company.

Projects are also defined by producing a product or meeting the objectives. A project that produces the product must meet all kinds of client requirements. The produced product must be justified by the client. The project is also required to satisfy certain objectives. The objectives can guide and motivate individual or groups to perform well in their assigned tasks.

In general, software projects have an objective-driven stage that recommends the new system to meet identified requirements and a project-driven stage to actually develop the product.

1.6 Setting Objectives

To develop a successful project, the project manager and the team members must be aware of the factors that lead them to success. There must be well-defined objectives accepted by all the people involved in the development process. A project authority must be identified to have an overall authority over the project. This authority is governed by a project steering committee also called as a project management board. Day-to-day activities must be reported to the steering committee by the project manager at regular intervals. Any changes to the defined objectives can be done only by the steering committee.

An effective objective's scope for an individual must be within the individual's control. Objectives can be broken down into goals or sub-objectives in order to achieve them. **SMART** technique is used for a well-defined objective.

- **Specific** ; Concrete and well-defined; Up to the point.
- **Measurable** : measures of effectiveness.
- **Achievable** ; power within the individual or the group.
- **Relevant** : satisfy the purpose of the project.
- **Time-oriented** : time limit for successful achievement of the project.

The objectives are met only when the system becomes operational. Performance measures deals the reliability of the operational system and predictive measures are done during the development of the project by measuring the effectiveness of the developing system.

1.7 Management Principles

- **Division of Work** - According to this principle the whole work is divided into small tasks. The specialization of the workforce according to the skills of a person, creating specific personal and professional development within the labour force and therefore

increasing productivity; leads to specialization which increases the efficiency of labour.

- **Authority and Responsibility** - This is the issue of commands followed by responsibility for their consequences. Authority means the right of a superior to give enhance order to his subordinates; responsibility means obligation for performance.
- **Discipline** - It is obedience, proper conduct in relation to others, respect of authority, etc. It is essential for the smooth functioning of all organizations.
- **Unity of Command** - This principle states that each subordinate should receive orders and be accountable to one and only one superior. If an employee receives orders from more than one superior, it is likely to create confusion and conflict.
- **Unity of Direction** - All related activities should be put under one group, there should be one plan of action for them, and they should be under the control of one manager.
- **Subordination of Individual Interest to Mutual Interest** - The management must put aside personal considerations and put company objectives firstly. Therefore the interests of goals of the organization must prevail over the personal interests of individuals.
- **Remuneration** - Workers must be paid sufficiently as this is a chief motivation of employees and therefore greatly influences productivity. The quantum and methods of remuneration payable should be fair, reasonable and rewarding of effort.
- **The Degree of Centralization** - The amount of power wielded with the central management depends on company size. Centralization implies the concentration of decision making authority at the top management.
- **Line of Authority/Scalar Chain** - This refers to the chain of superiors ranging from top management to the lowest rank. The principle suggests that there should be a clear line of authority from top to bottom linking all managers at all levels.
- **Order** - Social order ensures the fluid operation of a company through authoritative procedure. Material order ensures safety and efficiency in the workplace. Order should be acceptable and under the rules of the company.

- **Equity** - Employees must be treated kindly, and justice must be enacted to ensure a just workplace. Managers should be fair and impartial when dealing with employees, giving equal attention towards all employees.
- **Stability of Tenure of Personnel** - Stability of tenure of personnel is a principle stating that in order for an organization to run smoothly, personnel (especially managerial personnel) must not frequently enter and exit the organization.
- **Initiative** - Using the initiative of employees can add strength and new ideas to an organization. Initiative on the part of employees is a source of strength for organization because it provides new and better ideas. Employees are likely to take greater interest in the functioning of the organization.
- **Esprit de Corps/Team Spirit** - This refers to the need of managers to ensure and develop morale in the workplace; individually and communally. Team spirit helps develop an atmosphere of mutual trust and understanding. Team spirit helps to finish the task on time.

1.8 Management Control

The following are common types of management control.

- Structures. Organizational structures such as authority, roles, accountability, responsibility and separation of concerns.
- Objectives
- Performance Management.
- Task Assignment
- Setting Expectations
- Supervision.
- Measurements
- Monitoring

1.9 Project Portfolio Management

Project portfolio management provides an overview of all the projects that an organization is undertaking or is considering.

The concerns of project portfolio management include:

- Identifying which project proposals are worth implementation
- Assessing the amount of risk of failure that a potential project has
- Deciding how to share limited resources, including staff time and finance, between projects

The three key aspects of project portfolio management are

- Portfolio Definition
- Portfolio Management
- Portfolio Optimization

1.10 Cost Benefit Evaluation Technology

The Cost Benefit Evaluation techniques are

- Net profit
- Payback period
- Return on Investment
- Net Present Value
- Internal Rate of Return

Net Profit

- The difference between the total costs and the total income over the life of the project is calculated as net profit.
- Net profits do not involve the timing of the cash flows. When there are many projects, the net profit of preferable projects is done on selection criteria.
- Some projects incomes are returned only towards the end of the project. This is a major disadvantage which means that the investment must be funded for longer time.
- Estimates in distant future are less reliable than the short-term estimates which are more preferable.

Payback Period

- The time taken to break even or pay back the initial investment is the payback period.

- The project with the shortest payback period will be taken based on organizations that wish to minimize the time limit.
- The payback period is simple to calculate but sensitive to forecasting errors.
- The limitation of the payback period is that it ignores the overall profitability of the project.

Return on Investment

- The accounting rate of return or the return on investment compares the net profitability to the investment required.
- Return on Investment (ROI) is calculated using the given formulae;

$$ROI = \frac{\text{average annual profit}}{\text{total investment}} \quad ROI = \frac{\text{average annual profit}}{\text{total investment}} \times 100$$

- The ROI provides simple, easy to calculate the measure of return on capital.

Eg: The net profit of a project is Rs.30,000 and the total investment is Rs.100,000.

Calculate the ROI if the total period is taken as 3 years.

$$ROI = \frac{\text{average annual profit}}{\text{total investment}} \quad ROI = \frac{\text{average annual profit}}{\text{total investment}} \times 100$$

$$\begin{aligned} &= \frac{30,000}{100,000} = 30,000 \times \frac{1}{100,000} \times 100 \\ &= 10\% \end{aligned}$$

- The limitations of ROI is that it takes no account of the timing of the cash flows and does not bother about the compound interest.

Net Present Value

- Net present value is a project evaluation technique that is determined by the profitability of the project and the timing of the cash flows produced.

- The annual rate of return with respect to discounted future earnings is termed as the discount rate.
- The net present value of any future cash flow is calculated by the formulae:

$$\text{Present value} = \text{value in year } t / (1+r)^t$$

Where 'r' denotes the discount rate expressed as a decimal value,

't' represents the number of years of future cash flows.

- Net present value can also be calculated by multiplying the cash flow by the appropriate discount factor.
- NPV for a project is obtained by summing the discounted values and discounting each cash flows.
- The discount rates must be standard and it should reflect the interest rates as nominal with similar projects which is uncertain with NPV method.
- Using NPV, the measure of profitability of comparable projects is not directly concerned with earnings from other investments which are quoted as a percentage interest rate.

Internal Rate of Return

- The limitation of NPV is over come by the internal rate of return method. This method provides a profitability measure as a percentage return that is directly compared with interest rates.
- The percentage discount rate which produces an NPV of zero is calculated by IRR.
- A spreadsheet or a small computer program can be used to calculate the IRR is a convenient and useful measure of value of a project.
- A project with an IR greater than the current interest rates provides better return rate than lending from a bank.
- The limitation of IRR is that it does not indicate the absolute size of the return value.
- A total evaluation takes into account the problems of cash flow funding where as a project's IRR indicates that the profitable project future earnings are less reliable than investing with a bank.

1.11 Risk Evaluation

- Risk is associated with almost every project. Risk can become an important factor when the project is not able to meet its objectives.
- Every possible risk must be identified, analyzed and minimized during the development of the software system.

Risk Identification

- Every project evaluation involves risk handling issues.
- All possible risks are identified and must be quantified with their potential measures of evaluation.
- A project risk matrix can be implemented in creating a checklist of all possible risks and classify them based on their relative importance.
- The risk matrix contains values of high, medium and low based on their likelihood.
- Some factors classified in the risk project matrix contains, delivery of software, development budget exceeded limit, estimation of maintenance costs, response time targets and so on.

Risk Ranking

- Based on the risk identified, ranking can be established for projects.
- Evaluating projects based on the risk project matrix gives a clear picture of how to rank the different risks that occurs in projects.
- Risk ranking involves giving scores to projects based on priorities defined for each risk in the project.

NPV and Risk

- Risky projects always have a higher discount rate for the net present value calculation.
- The risk level may be very high for a specific project due to rise in NPV value. So based on risk scores, projects are classified as high, medium and low level.
- It is better to have an additional risk premium factor to have an consistent method in developing the project.

- Discounted cash flow techniques can be used to evaluate the net present value of future cash flow taken into account the interest rates and uncertainty.

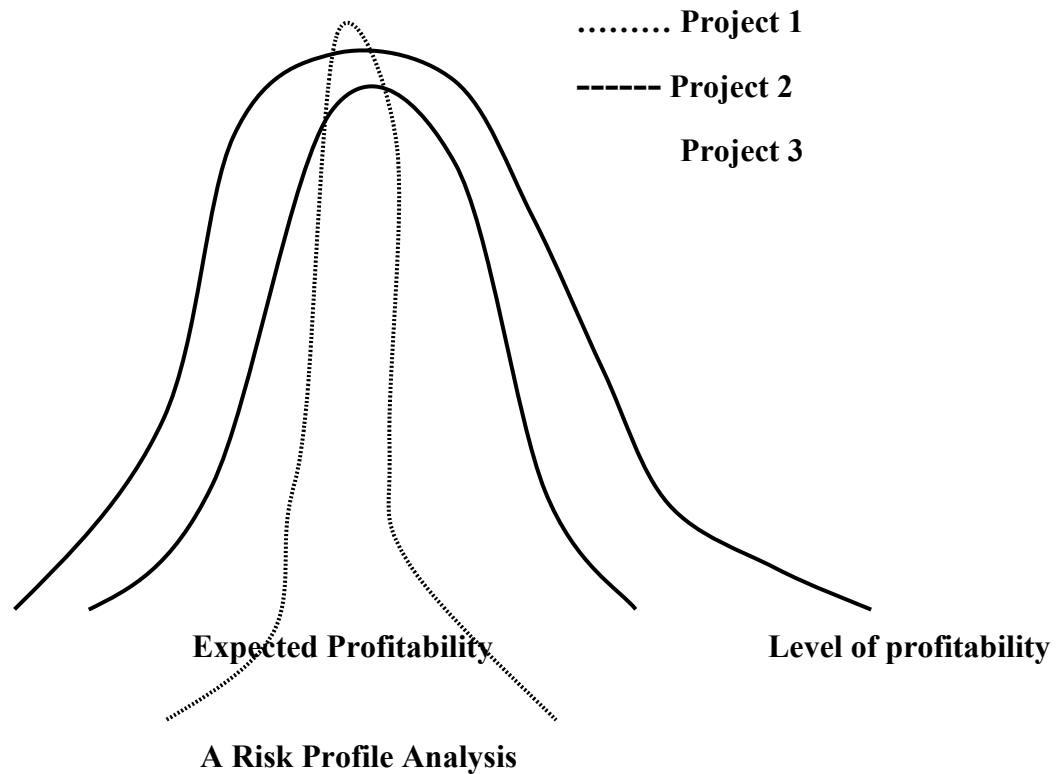
Risk in Cost benefit analysis

- Cost-benefit analysis focuses on the estimated cost defined for the project compared with the actual costs incurred in the development process.
- Evaluation of risk involves the possible outcomes of the project by estimating the probability of occurring.
- A group of cash flow forecasts associated with each probability of occurring can be defined and the value summarizes the cost or benefit of each possible outcome weighted with its relative probability.
- Basically, cost-benefit analysis is done for evaluation of larger projects which are subject to uncertainty.
- It is most appropriate to evaluate a portfolio of projects to determine the overall profitability.

Risk Profile Analysis

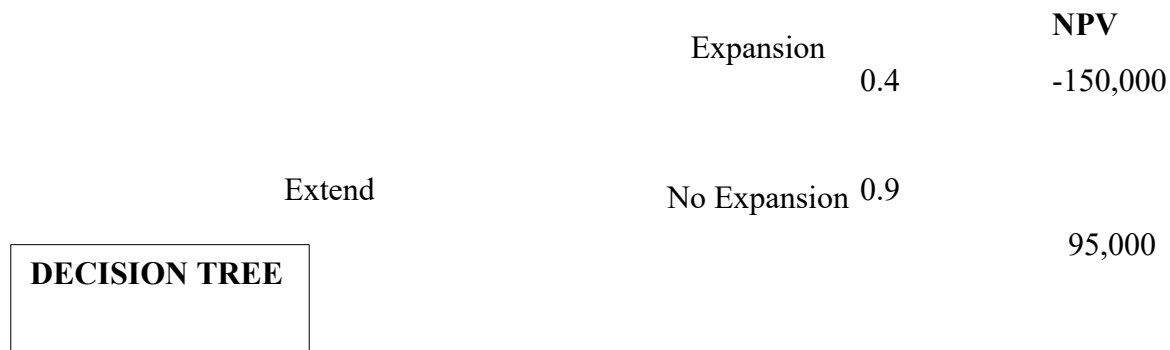
- Risk profiles are constructed using sensitivity analysis which involves the sensitivity factors that affect the project costs or benefits.
- For example, the original estimate of a project was calculated with plus or minus 5% of risk, then calculating the expected costs and benefits for each of the estimating factor results in evaluating the sensitivity of the project.
- Sensitivity analysis identifies the factors that yields a success to the project and decide about whether to carry on with the project or lay off.
- The sensitivity analysis takes into account every risk factor, and evaluates on the possible chances of a particular outcome of the project.
- Monte Carlo simulation tool is used to find out the number of possible chances of specific project.
- A sample risk analysis profile is depicted in the figure below:

- Consider three projects 1, 2 and 3, the figure describes that project 1 is very far from the expected value compared to project 2. Project 2 exhibits a larger variance where as project 3 represent a skewed distribution. Project 3 can attain the profitability than expected but it can go worse too.



Risk handling using Decision trees

- It is better to reject projects than working with risky ones.
- Decision trees is a tool which provides evaluation of project's expected outcomes and choosing between the alternative strategies.



Replace	Expansion	0.4	300,000
	No Expansion	0.9	-70,000

Any decision that is made will have a greater impact on the future profitability of the project.

- The analysis of a decision tree consists of evaluating the expected benefit of taking each path from a decision point.
- The expected value of each path is determined by the sum of the value of each possible outcome multiplied by its probability of occurrence.
- The figure illustrates the use of decision tree of when to extend the project or replace the existing system based on the NPV values defined.
- Decision trees are more advantageous because they will give a precise idea of modeling and analyzing the problems in the project.

1.12 Strategic program management

Programme Management Definition

A classical definition of programme management by D.C.Ferns, “**a group of projects** that are maintained in a coordinated way to gain benefits that would not be possible where the projects to be managed independently”.

Different forms of Programme Management

There are various forms of programme management that exist. They are

- Strategic programmes
- Business cycle programmes
- Infrastructure programmes
- Research and development programmes

Strategic Programmes

- Portfolio programme models define a strategic domain process within the organization.

- Group of projects can lead to single strategy.
- Organizations can be grouped together and every activity associated with each distinct project can be controlled and coordinated manner as a programme.

Business Cycle Programmes

- A project portfolio is a group of projects carried out under the sponsorship or management of an organization.
- Prioritizing projects must be based on decisions made by the project manager to handle them in different situations.
- If one project needs more resources than expected, expenses can be incorporated from other projects giving preference to the former one.
- Importance must be given to individual projects inside the portfolio.

Infrastructure Programmes

- Organizations differ in the way they exist. Some of them have distinct departments

while others have integrated systems.

- Each department might be unique in handling different information having distinct databases defined.
- A uniform infrastructure will allow sharing of applications between various departments which would help in the development process.

Research and Development Programmes

- Innovative companies develop new products that are too risky. If the new product fails in the market, it will be difficult to handle the situation.
- On the other hand, the new product becomes success, then there will be huge reap in the business organization.
- Certain development projects results in a good planned project. But projects that are too risky if successful yields more benefit than the innovative ones.

- A risk involved fluctuates in a innovative project. Research projects leading to new discovery results in technological revolution that affect the market.
- For example, internet and world wide web has helped in adopting innovative and research development programmes.

Strategic Programme Management

- A programme manager must possess these qualities:
 - Managing simultaneous projects inside the portfolio
 - Resources must be well understood
 - Utilization of resources must be attained
 - Optimal usage of specialist staff for specific tasks
- When portfolios of projects contribute to a common objective, it leads to strategic management.
- To have consistent and uniformity of projects, a business objective is defined to coordinate the project at a different level.
- Large organizations typically have a large and complicated organizational structure. For example, government department like OGC (Office of Government Commerce) has defined effective guidelines for the strategic programme development.

Creating a Programme

- The various phases involved in creating a programme are defined as:
 - Creation of programme mandate
 - Programme brief
 - Vision statement
 - Blueprint of programme
 - Programme portfolio

Creation of programme mandate

- A programme mandate contains a formal document containing:
 - New services the programme delivers

- Benefit of organization by new services
 - Meeting the corporate goals and other initiatives
- A programme director is nominated within the organization to provide an initial leadership for the programme. The programme director must be from the sponsoring group which has already identified the need for the programme.

Programme brief

- A programme brief defines the feasibility study of the programme. It includes:
- Preliminary vision statement highlighting the capacity of the organization.
 - Benefits generated from the programme
 - Risks and other issues involved
 - Estimated cost, effort and time limit for completion

Vision statement

- The vision statement describing the sponsoring group with a more detailed planning process.
- To govern the day to day responsibilities a programme manager is appointed from within the project management team for running the programme.
- Programme manager along with the project development team analyzes the vision statement and formulates a refined plan for implementing the process.

Blueprint of programme

- The description of the vision statement and the changes that have been made to the structure and the operations are represented in the blueprint.
- A blueprint must emphasize on:
- Requirement of business models for the new process
 - Staff requirement by the organization
 - Resources requirements
 - Data and information requirements
 - Cost, effort, performance and service level requirements

Programme portfolio

- Initially, a list of projects are created along with its objectives to create a programme portfolio.
- An outline schedule of the entire development process is presented by the sponsoring group with all estimation factors.
- Groups are identified with similar interest and drawn out as a stakeholder map.
- A communication strategy and plan shows the appropriate information flow between stakeholders.

The preliminary plan produces the project portfolio, estimation of costs, expected benefits, risks identified and the resources needed

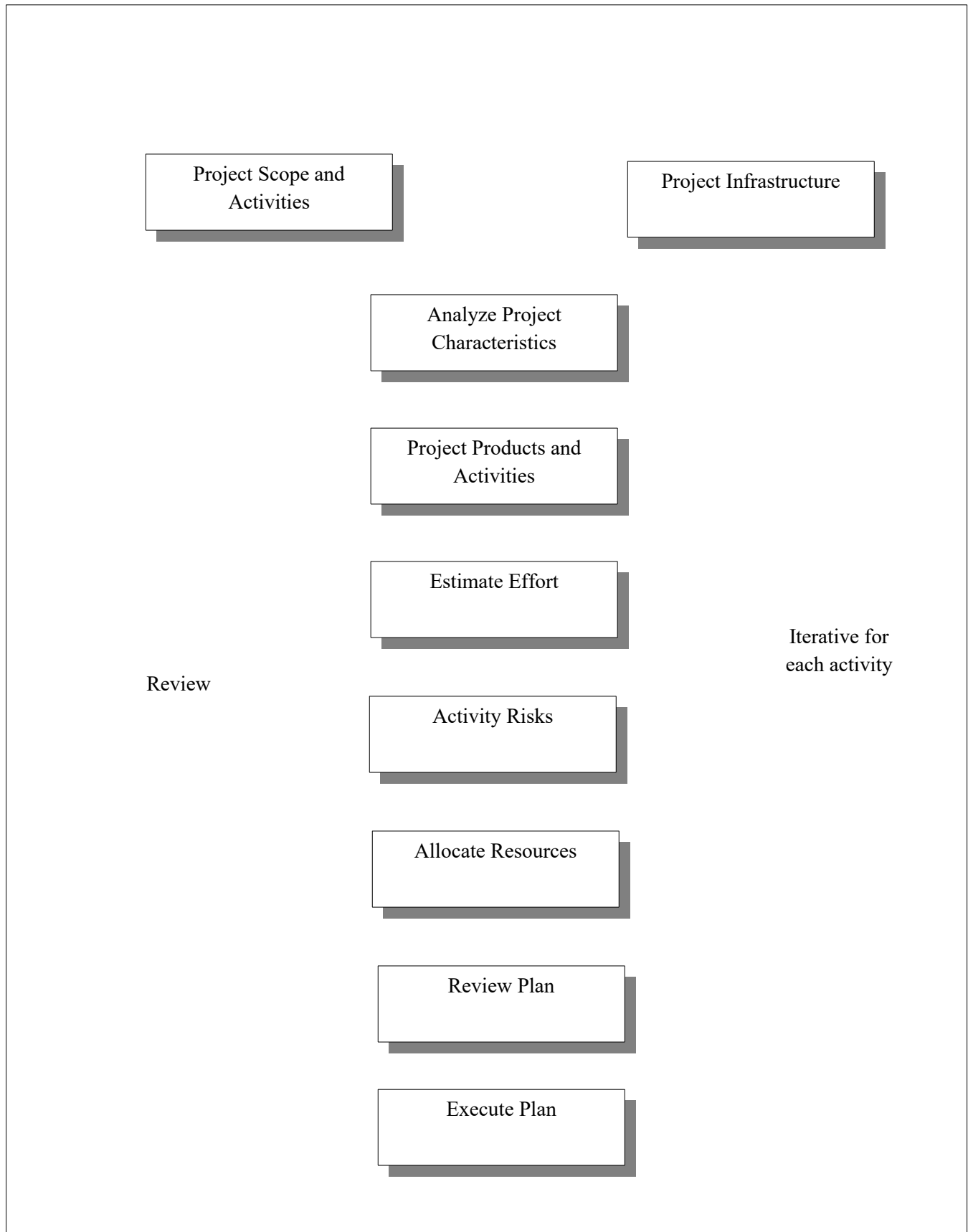
1.13 Stepwise Project Planning

Outline of Step Wise Project Planning

The framework of basic steps in project planning illustrates the various activities involved in the development process.

An outline of Step Wise planning is listed below:

- Selecting project
- Project scope & objectives
- Project infrastructure
- Analyze project characteristics
- Project products and activities
- Estimation effort
- Activity risks
- Allocate resources
- Review plan
- Execute plan



Step 0: Selecting Project

- This is the initial step which starts well outside the project planning process.
- Feasibility study of the project helps in choosing the appropriate one.
- Strategic planning process helps in evaluating the metrics of selecting the project.
- Different methodologies are inevitable, stemming directly from the questions of what constitutes a methodology and what are a methodology's underlying principles.
- Projects differ according to size, composition, priorities, and criticality.
- The people on a project have different biases based on their experiences, principles, and fears.
- These issues combine so that, what is optimal differs across projects.
- Projects are undertaken to produce a product or a service for various reasons.
- This includes factors like market share, financial benefits, return on investment, customer retention and loyalty, and public perceptions.
- Organizations might receive several projects at a time. They have to select the best among the received projects request.
- They make decisions based on the best information they have about a particular project at a given point of time when selecting the project.

Step 1: Project Scope and Objectives

- Every stakeholder involved in the project must agree on the objectives defined in determining the success of the project.
- Scope statements may take many forms depending on the type of project being implemented and the nature of the organization.
- The scope statement details the project deliverables and describes the major objectives.
- The objectives should include measurable success criteria for the project.
- The Scope Statement should be written before the Statement of work and it should capture, in very broad terms, the product of the project, for example, *"developing a software based system to capture and track orders for software."*

- The Scope Statement should also include the list of users using the product, as well as the features in the resulting product.
- As a baseline scope statements should contain:
 - The project name
 - The project charter
 - The project owner, sponsors, and stakeholders
 - The problem statement
 - The project goals and objectives
 - The project requirements
 - The project deliverables
 - The project non-goals
 - Milestones
 - Cost estimates
- In more project oriented organizations the scope statement may also contain these and other sections:
 - Project Scope Management Plan
 - Approved change requests
 - Project assumptions and risks
 - Project acceptance criteria
- The project objectives are identified and practical measures are analyzed in achieving them
- A project authority must be identified to have an overall authority over the project.
- Identify different stakeholders involved in the development of the project.
- Changes in the objectives must be in a controlled manner.
- Interaction and communication among all parties must be straight forward.

Step 2: Project Infrastructure

- Project Infrastructure refers to the organizational structure, processes, tools, techniques and training an organisation puts in place to make projects more successful.
- Organisational Structure – Organisational structure including such support mechanisms as project management office, project recruiting function, financial monitoring area etc. It also covers lines of communication and escalation.
- Processes – Typically methodologies, checklists and guidelines
- Tools – Software and templates
- Techniques – Repeatable processes such as kick off meetings, PIRs, analysis techniques, etc.
- Training – Formal and informal training and reference documentation
- Organization must give priorities for multiple projects to be carried out.
- Strategic decisions must be documented within the strategic plan in identifying the relationship between multiple projects.
- Change control must be implemented without affecting the original objectives.
- Configuration and procedural standards are defined for quality checks at regular intervals of the SDLC process and documented in separate manual.
- Measurement programme determines the control policy and monitors the progress of the project.
- Project manager must have an overall control of any project planning and control standards adopted.
- Project leader takes the responsibility of building the project team as an organized, well-built and effective one yielding excellent results.
- Team members must work together as a team and resolve conflicts.

Step 3: Analyze Project Characteristics

- The project is categorized as either product-driven or an objective-driven.
- A project has several characteristics:
 - * Projects are unique.
 - * Projects are temporary in nature and have a definite beginning and ending date.

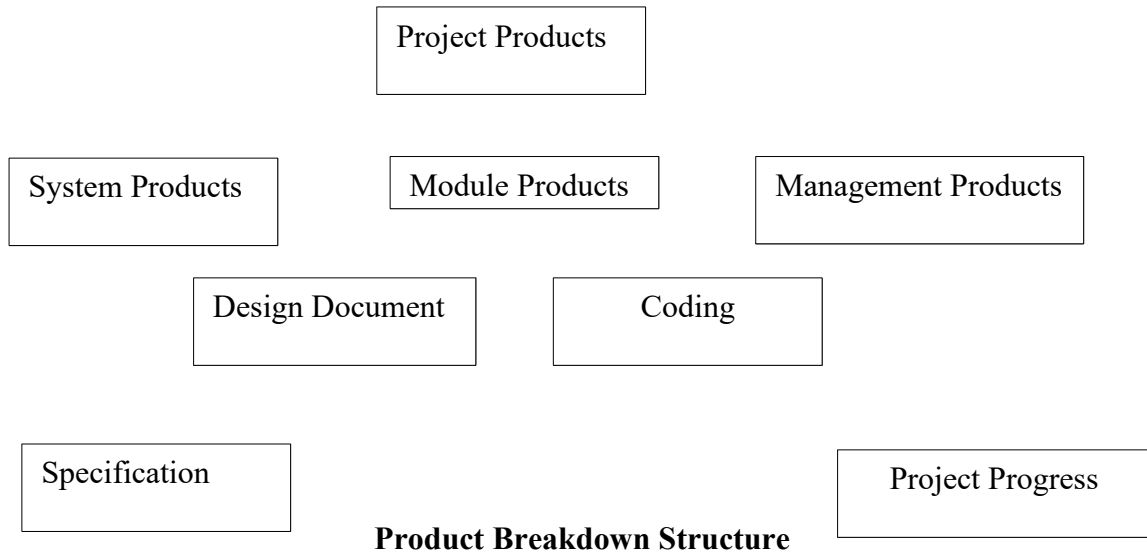
* Projects are completed when the project goals are achieved or it's determined the project is no longer viable.

* A successful project is one that meets or exceeds the expectations of your stakeholders.

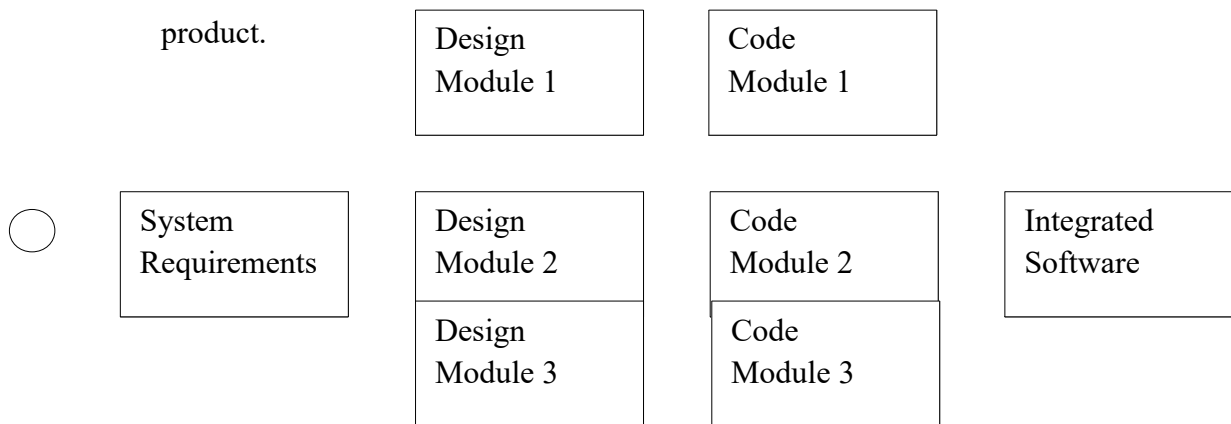
- As the system is developed, the product is driven out of the defined objectives.
- The project must be analyzed based on its quality requirements.
- Projects are prone to higher risk which needs to be handled without affecting the product created.
- In implementing the product, user requirements are given due importance.
- Appropriate methodology and SDLC process must be chosen to suit the current product.
- Review the overall resource estimates.

Step 4: Project Products and Activities

- Identify the project deliverables i.e. the end product that has to be given over to the client.
- Some products are identified as intermediate products during the creation of deliverables.
- Project products can be System products, module products or management products.
- Technical products include training materials and operating instructions in managing the quality of the project.
- Describe the project products into components and sub-components related to individual modules in each step.
- Every activity must be carried out for each stage of the development process.
- Management products include progress of the project that is developed.
- Product descriptions contain the identity, purpose, derivation, composition, form, relevant standard and the quality criteria that apply.
- Not all products are independent. Some products depend on other products for their creation.



- Product flow diagram represents the flow of the product being developed.
- Product instances must be recognized when a product is related to more than one product.



Sample Activity Network

- An activity network is created for generating the product that depends on another product describing every task associated with it.
- Sequencing of activities minimizes the overall duration for the project.

- For a complex project, the entire project can be divided into stages and checkpoints can be formulated at each specific stage for compatibility.
- Milestones represents the completion of important stages of the project.

Step 5: Estimating Effort

- The effort estimation for the staff required, the probable duration and the non-staff resources needed for every activity is determined.
- These estimates depend on the type of the activity.
- Effort is the amount of work that has to be done.
- Software development efforts estimation is the process of predicting the most realistic use of effort required to develop or maintain software based on incomplete, uncertain and/or noisy input.
- Effort estimates may be used as input to project plans, iteration plans, budgets, investment analyses, pricing processes and bidding rounds.
- Elapsed time is the time between the start and end of a task.
- With all the activities defined, the overall duration of the project can be calculated using the activity network.
- For longer activities it will be difficult to control the project over estimating factors.
- There are many ways of categorizing estimation approaches. The top level categories are the following:
 - Expert estimation: The quantification step, i.e., the step where the estimate is produced based on judgmental processes.
 - Formal estimation model: The quantification step is based on mechanical processes, e.g., the use of a formula derived from historical data.
 - Combination-based estimation: The quantification step is based on a judgmental or mechanical combination of estimates from different sources.
- The uncertainty of an effort estimate can be described through a prediction interval (PI). An effort PI is based on a stated certainty level and contains a minimum and a maximum effort value.

- The most common measures of the average estimation accuracy is the MMRE (Mean Magnitude of Relative Error), where MRE is defined as:

$$\text{MRE} = |\text{actual effort} - \text{estimated effort}| / |\text{actual effort}|$$

- Psychological factors potentially explaining the strong tendency towards over-optimistic effort estimates that need to be dealt with to increase accuracy of effort estimates.
- These factors are essential even when using formal estimation models, because much of the input to these models is judgment-based.
- Factors that have been demonstrated to be important are: Wishful thinking, anchoring, planning fallacy and cognitive dissonance.
- The psychological factors found in work by Jorgensen and Grimstad describes,
 - It's easy to estimate what you know.
 - It's hard to estimate what you know you don't know.
 - It's very hard to estimate things that you don't know you don't know.

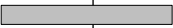






Step 6: Identify Activity Risks

- Activity based risks are identified for every activity based on number of assumptions.
- Risk planning reduces the impact of identified risks.
- To materialize the risk, contingency plans are specified.
- New activities can reduce risks to a certain extent when there is change in plans.
- Risks fall into three broad categories — controllable known, uncontrollable known and unknown.
- The former two, are those risks happen before they can determine how to manage them. This is done using root cause analysis.
- As the name implies its goal is to look for the root cause on the problem and solve it at that point.
- The four ways of handling risk are:
 - **Avoidance** - Take action to avoid the risk
 - **Mitigation** - Define actions to take when the risk occurs
 - **Transfer** - Have someone else handle the risk i.e. insurance
 - **Acceptance** - Identify the risk as acceptable and let it happen.

- Determining which option to chose is primarily financial, but schedule and manpower may be involved.
- As a tool, a number of "checklist" opinions for looking at each of these options.
- Contingency planning is briefly discussed for scope, resource and schedule.

Step 7: Allocate Resources

- Resource allocation is used to assign the available resources in an economic way. It is part of resource management. In project management, resource allocation is the scheduling of activities and the resources required by those activities while taking into consideration both the resource availability and the project time.
- Staff needed and available are identified for each activity and allocated their respective tasks.

Tasks / Months	JAN	FEB	MAR	APR	MAY
System requirements					
Devise Integration test cases					
Design module 1					
Code module 1					
Design module 2					
Code module 2					
Integrated software					

Gantt chart showing staff tasks

- Staff priority list is generated based on the task allotted to them because some staffs are used for more than one task.

- A Gantt chart pictorially represents when activities have to take place and which one has to be executed at the same time.
- The chart represents when staff will be carrying out the tasks in each month. It also shows staff involved in more than one task.
- When allocating resources the constraints associated is estimated and included in the overall cost.

Step 8: Review Plan

- When a task is completed it leads to the quality review. These quality checks have to be passed before the activity is completely signed-off.
- Every plan has to be documented and all stakeholders must have agreed to all constraints and understand the project.
- There are some steps involved in project plan review.
 - **Define the problem** This activity provides the background for decisions about the scope and focus of the Project Review. Here are some simple questions the Project Review Team can ask themselves before creating a plan for the project. Use our Planning Tool to capture the background on your project.
 - ❖ What, if any, review work has already been done?
 - ❖ What is the problem we are trying to solve?
 - ❖ What would success look like?
 - ❖ Scope the Project. How big was it? How long did it take? How many people were involved?
 - ❖ What is the investment the team would like to make?
 - **Determine the focus** : The focus of the Project Review is the question that the team will ask themselves as they investigate the events that occurred during the project. This is the fundamental question that will guide the decisions that the team will make while planning the Project Review. It is always stated as a question. A commonly used question that project teams ask is:
 - ❖ What are the root causes of events that determined or impacted resources, schedule, or quality?

- **Select the appropriate tools** : Now that the scope, the goal and the problem are known, the data set needed for the project review are identified along with the various activities that will be used.
- **Identify the participants** : The Project Review Leadership Team guides the Postmortem effort. As a group they determine the focus of the investigation, select the tools that will be used, review the output from each step, decide who should participate in each activity, and are responsible for reporting lessons learned and recommendations for action. The Project Review Team usually consists of the movers and shakers that drove the project or event. They work together to manage the Project Review process. The team should consist of folks most intimate with the project including any of the following representatives:
 - ❖ Project Managers
 - ❖ Product Managers
 - ❖ Development Leads
 - ❖ Quality Leads
 - ❖ Content Experts
 - ❖ Customer Support Leads
 - ❖ Management
- **Document the review plan** : The project review template can be used so that everyone responsible for implementation has a copy of the plan.

Step 9: Execute Plan

- Finally, the execution of the project is drawn with each specified activity as it is approached.
- Detailed planning of later stages is necessary because more information will be available than the start stage.
- Project planning and execution becomes an iterative process where as each activity which is to be carried out approaches, they should be reviewed in detail.

UNIT II - PROJECT LIFE CYCLE AND EFFORT ESTIMATION

2.1 Software Process and Process Models

A Software product development process usually starts when a request for the product is received from the customer.

- Starting from the inception stage:
 - A product undergoes a series of transformations through a few identifiable stages
 - Until it is fully developed and released to the customer.
- After release:
 - the product is used by the customer and during this time the product needs to be maintained for fixing bugs and enhancing functionalities. This stage is called Maintenance stage.
- This set of identifiable stages through which a product transits from inception to retirement form the life cycle of the product.
- Life cycle model (also called a process model) is a graphical or textual representation of its life cycle.

2.2 Choice of Process Models

The no. of inter related activities to create a final product can be organized in different ways and we can call these Process Models.

A software process model is a simplified representation of a software process. Each model represents a process from a specific perspective. These generic models are abstractions of the process that can be used to explain different approaches to the software development.

Any software process must include the following four activities:

1. **Software specification** (or requirements engineering): Define the main functionalities of the software and the constraints around them.
2. **Software design and implementation**: The software is to be designed and programmed.
3. **Software verification and validation**: The software must conform to its specification and meets the customer needs.

4. **Software evolution** (software maintenance): The software is being modified to meet customer and market requirements changes.

The various Process Models are

Water Fall Model

Spiral Model

Prototype Model

Incremental Delivery

2.2.1 Water fall model

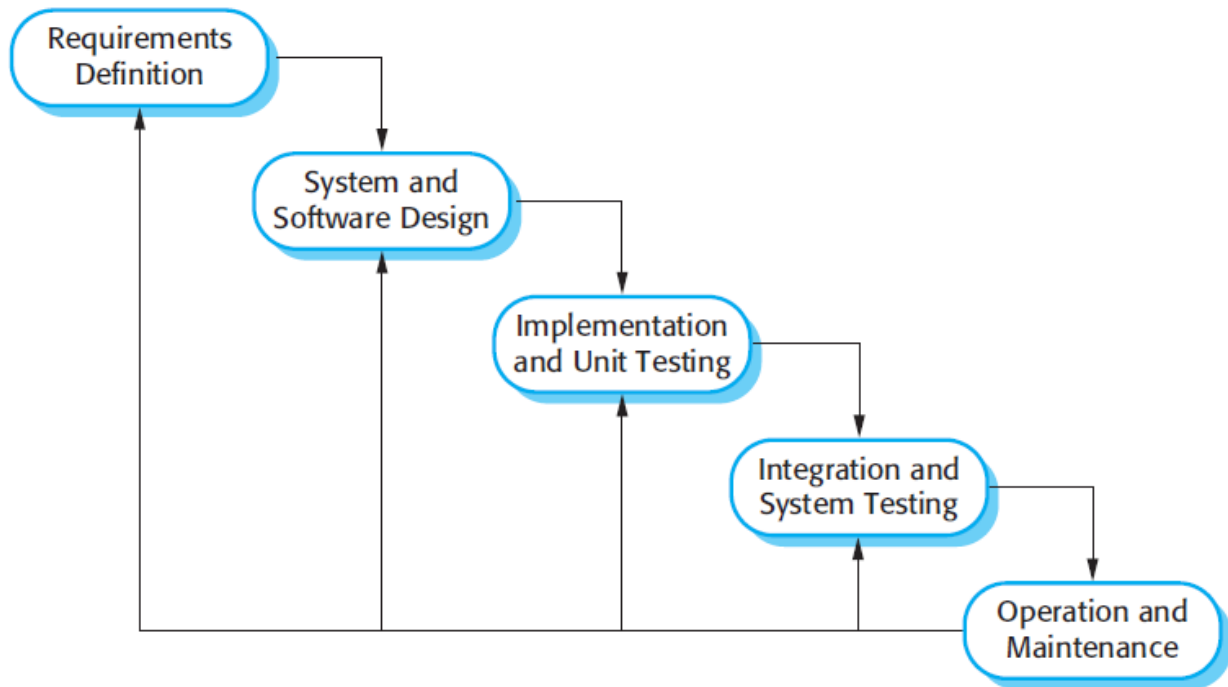
The waterfall model is a sequential approach, where each fundamental activity of a process represented as a separate phase, arranged in linear order.

In the waterfall model, you must plan and schedule all of the activities before starting working on them (plan-driven process).

Plan-driven process is a process where all the activities are planned first, and the progress is measured against the plan. While the agile process, planning is incremental and it's easier to change the process to reflect requirement changes.

The phases of the waterfall model are:

- Requirements
- Design
- Implementation
- Testing
- Maintenance



In principle, the result of each phase is one or more documents that should be approved and the next phase shouldn't be started until the previous phase has completely been finished.

In practice, however, these phases overlap and feed information to each other. For example, during design, problems with requirements can be identified, and during coding, some of the design problems can be found, etc.

The software process therefore is not a simple linear but involves feedback from one phase to another. So, documents produced in each phase may then have to be modified to reflect the changes made.

2.2.2 Spiral Model

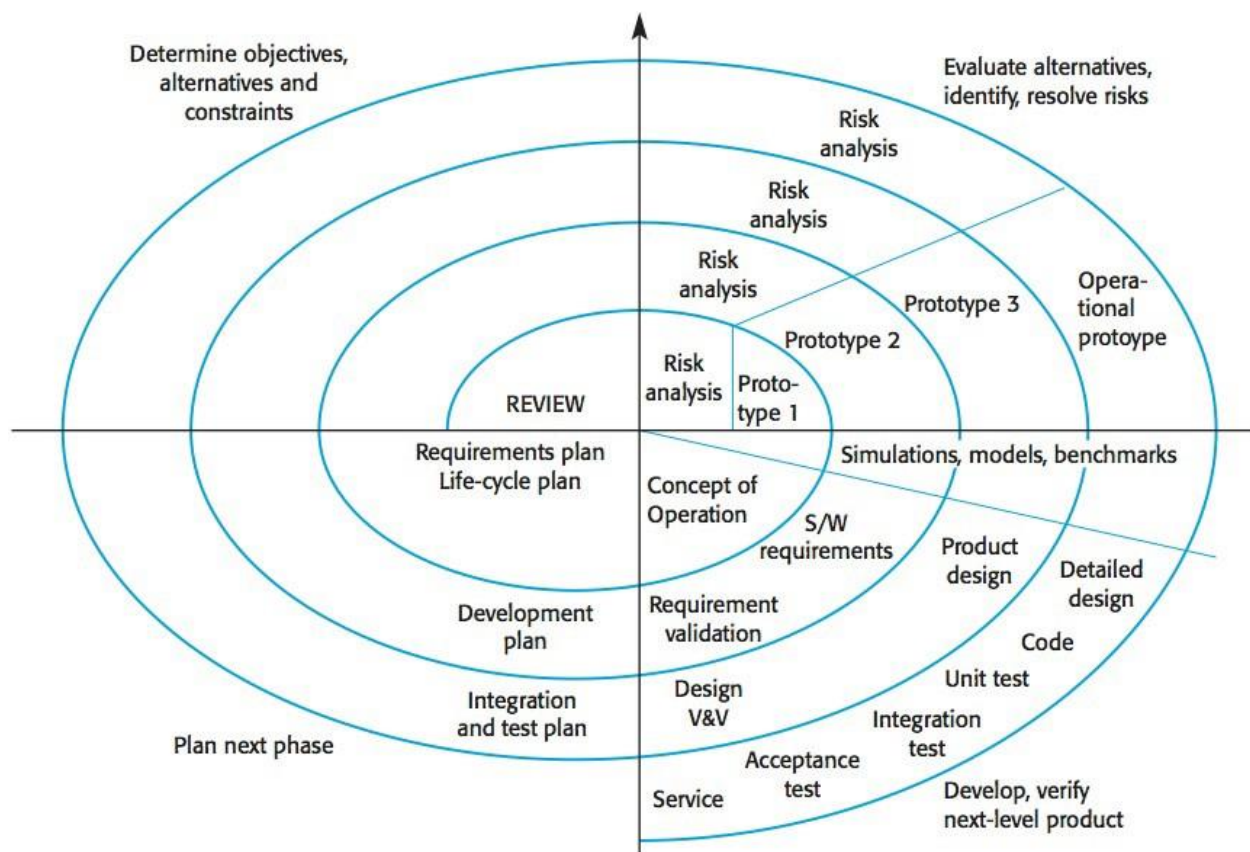
The spiral model is similar to the incremental model, with more emphasis placed on risk analysis. The spiral model has four phases: Planning, Risk Analysis, Engineering and Evaluation. A software project repeatedly passes through these phases in iterations (called Spirals in this model). The baseline spiral, starting in the planning phase, requirements is gathered and risk is assessed. Each subsequent spiral builds on the baseline spiral. It is one of the software development models like Waterfall, Agile, V-Model.

Planning Phase: Requirements are gathered during the planning phase. Requirements like 'BRS' that is 'Business Requirement Specifications' and 'SRS' that is 'System Requirement specifications'.

Risk Analysis: In the **risk analysis phase**, a process is undertaken to identify risk and alternate solutions. A prototype is produced at the end of the risk analysis phase. If any risk is found during the risk analysis then alternate solutions are suggested and implemented.

Engineering Phase: In this phase software is **developed**, along with **testing** at the end of the phase. Hence in this phase the development and testing is done.

Evaluation phase: This phase allows the customer to evaluate the output of the project to date before the project continues to the next spiral.



Advantages of Spiral model:

- High amount of risk analysis hence, avoidance of Risk is enhanced.
- Good for large and mission-critical projects.
- Strong approval and documentation control.
- Additional Functionality can be added at a later date.
- Software is produced early in the software life cycle.

Disadvantages of Spiral model:

- Can be a costly model to use.
- Risk analysis requires highly specific expertise.
- Project's success is highly dependent on the risk analysis phase.
- Doesn't work well for smaller projects.

2.2.3 Software Prototyping

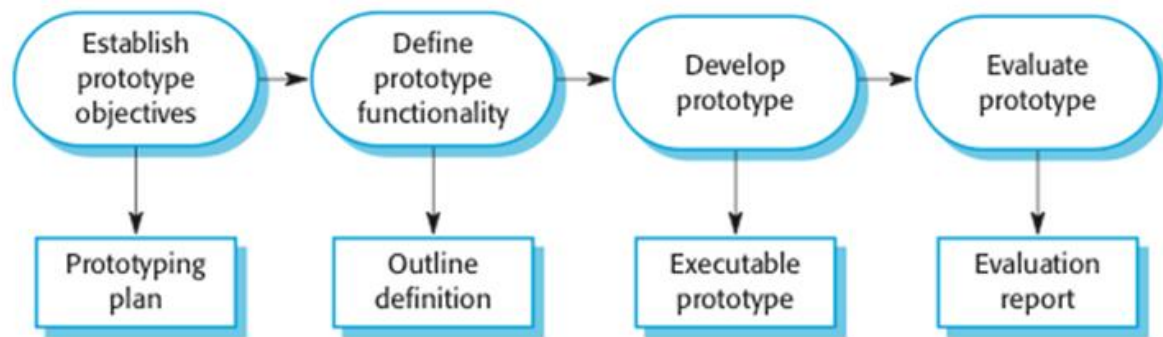
A prototype is a version of a system or part of the system that's developed quickly to check the customer's requirements or feasibility of some design decisions. So, a prototype is useful when a customer or developer is not sure of the requirements, or of algorithms, efficiency, business rules, response time, etc.

In prototyping, the client is involved throughout the development process, which increases the likelihood of client acceptance of the final implementation. While some prototypes are developed with the expectation that they will be discarded, it is possible in some cases to evolve from prototype to working system.

A software prototype can be used:

[1] In the **requirements engineering**, a prototype can help with the elicitation and validation of system requirements. It allows the users to experiment with the system, and so, refine the requirements. They may get new ideas for requirements, and find areas of strength and weakness in the software. Furthermore, as the prototype is developed, it may reveal errors and in the requirements. The specification may be then modified to reflect the changes.

[2] In the **system design**, a prototype can help to carry out design experiments to check the feasibility of a proposed design. For example, a database design may be prototype-d and tested to check it supports efficient data access for the most common user queries.



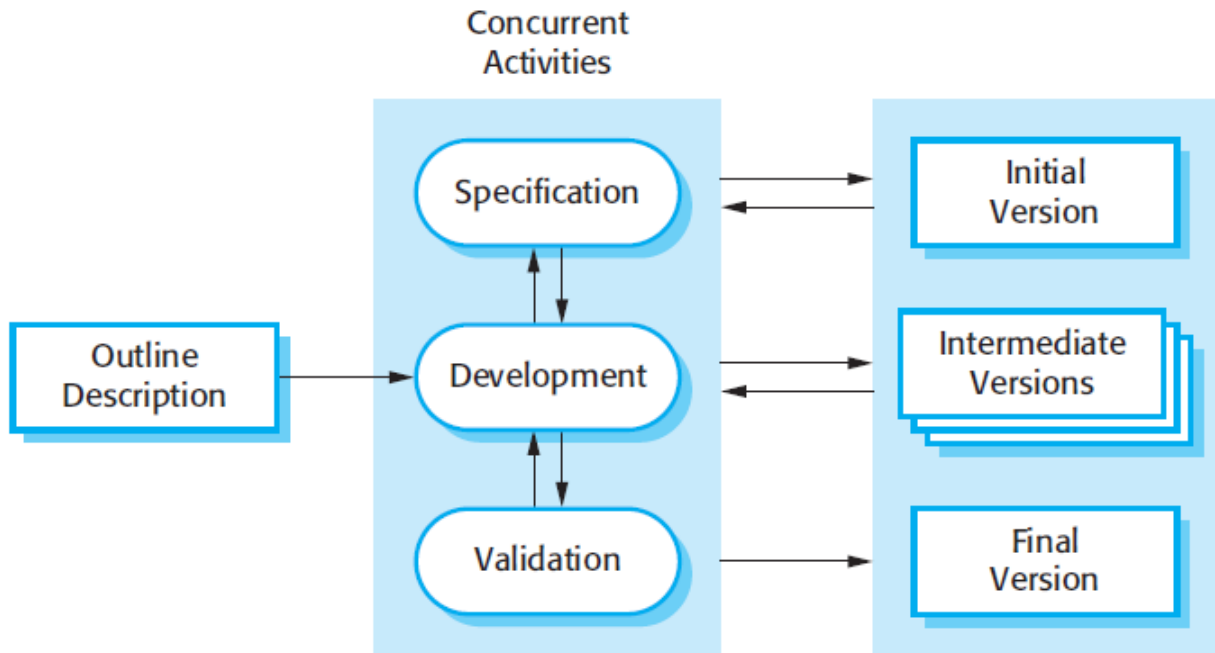
The phases of a prototype are:

1. **Establish objectives:** The objectives of the prototype should be made explicit from the start of the process. Is it to validate system requirements, or demonstrate feasibility, etc.
2. **Define prototype functionality:** Decide what are the inputs and the expected output from a prototype. To reduce the prototyping costs and accelerate the delivery schedule, you may ignore some functionality, such as response time and memory utilization unless they are relevant to the objective of the prototype.
3. **Develop the prototype:** The initial prototype is developed that includes only user interfaces.
4. **Evaluate the prototype:** Once the users are trained to use the prototype, they then discover requirements errors. Using the feedback both the specifications and the prototype can be improved. If changes are introduced, then a repeat of steps 3 and 4 may be needed.

Prototyping is not a standalone, complete development methodology, but rather an approach to be used in the context of a full methodology (such as incremental, spiral, etc).

2.3 Incremental Delivery

Incremental development is based on the idea of developing an initial implementation, exposing this to user feedback, and evolving it through several versions until an acceptable system has been developed. The activities of a process are not separated but interleaved with feedback involved across those activities.



Each system increment reflects a piece of the functionality that is needed by the customer. Generally, the early increments of the system should include the most important or most urgently required functionality.

This means that the customer can evaluate the system at early stage in the development to see if it delivers what's required. If not, then only the current increment has to be changed and, possibly, new functionality defined for later increments.

Incremental Vs Waterfall Model

Incremental software development is better than a waterfall approach for most business, e-commerce, and personal systems.

By developing the software incrementally, it is cheaper and easier to make changes in the software as it is being developed.

Compared to the waterfall model, incremental development has three important benefits:

1. The **cost of accommodating changing** customer requirements is reduced. The amount of analysis and documentation that has to be redone is much less than that's required with waterfall model.
2. It's easier to get **customer feedback** on the work done during development than when the system is fully developed, tested, and delivered.
3. More **rapid delivery** of useful software is possible even if all the functionality hasn't been included. Customers are able to use and gain value from the software earlier than it's possible with the waterfall model.

2.4 Rapid Application Development

RAD model is Rapid Application Development model. It is a type of incremental model. In RAD model the components or functions are developed in parallel as if they were mini projects. The developments are time boxed, delivered and then assembled into a working prototype. This can quickly give the customer something to see and use and to provide feedback regarding the delivery and their requirements.

The phases in the rapid application development (RAD) model are:

Business modeling: The information flow is identified between various business functions.

Data modeling: Information gathered from business modeling is used to define data objects that are needed for the business.

Process modeling: Data objects defined in data modeling are converted to achieve the business information flow to achieve some specific business objective. Description are identified and created for CRUD of data objects.

Application generation: Automated tools are used to convert process models into code and the actual system.

Testing and turnover: Test new components and all the interfaces.

Advantages of the RAD model:

- Reduced development time.
- Increases reusability of components
- Quick initial reviews occur

- Encourages customer feedback
- Integration from very beginning solves a lot of integration issues.

Disadvantages of RAD model:

- Depends on strong team and individual performances for identifying business requirements.
- Only system that can be modularized can be built using RAD
- Requires highly skilled developers/designers.
- High dependency on modeling skills
- Inapplicable to cheaper projects as cost of modeling and automated code generation is very high.

2.5 Agile Methods

The Agile methodology derives from a namesake manifesto, which advanced ideas that were developed to counter the more convoluted methods that pervaded the software development world despite being notoriously inefficient and counterproductive. Promoting similar methods to Lean, the key principles of Agile are as follows:

- Satisfying customers is of foremost importance
- Develop projects with inspired contributors
- Interactions are best when done in person
- Software that works is a measure of progress
- Reflect and adapt on an ongoing basis

Additionally, the four core values of Agile are as follows:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Regardless of which Agile methodology a team adopts, the benefits cannot be fully realized without the commitment of everyone involved.

2.5.1 Extreme Programming

One of the foremost Agile methodologies is called Extreme Programming (XP), which involves a high degree of participation between two parties in the software exchange: customers

and developers. The former inspires further development by emphasizing the most useful features of a given software product through testimonials. The developers, in turn, base each successive set of software upgrades on this feedback while continuing to test new innovations every few weeks.

XP has its share of pros and cons. On the upside, this Agile methodology involves a high level of collaboration and a minimum of up-front documentation. It's an efficient and persistent delivery model. However, the methodology also requires a great level of discipline, as well as plenty of involvement from people beyond the world of information technology. Furthermore, in order for the best results, advanced XP proficiency is vital on the part of every team member.

Extreme programming (XP) is a software development methodology which is intended to improve software quality and responsiveness to changing customer requirements. As a type of agile software development, it advocates frequent "releases" in short development cycles, which is intended to improve productivity and introduce checkpoints at which new customer requirements can be adopted.

Other elements of extreme programming include: programming in pairs or doing extensive code review, unit testing of all code, avoiding programming of features until they are actually needed, a flat management structure, code simplicity and clarity, expecting changes in the customer's requirements as time passes and the problem is better understood, and frequent communication with the customer and among programmers. The methodology takes its name from the idea that the beneficial elements of traditional software engineering practices are taken to "extreme" levels. As an example, code reviews are considered a beneficial practice; taken to the extreme, code can be reviewed *continuously*, i.e. the practice of pair programming.

Activities

XP describes four basic activities that are performed within the software development process: coding, testing, listening, and designing. Each of those activities is described below.

Coding

The advocates of XP argue that the only truly important product of the system development process is code – software instructions that a computer can interpret. Without code, there is no working product.

Coding can also be used to figure out the most suitable solution. Coding can also help to communicate thoughts about programming problems. A programmer dealing with a complex programming problem, or finding it hard to explain the solution to fellow programmers, might

code it in a simplified manner and use the code to demonstrate what he or she means. Code, say the proponents of this position, is always clear and concise and cannot be interpreted in more than one way. Other programmers can give feedback on this code by also coding their thoughts.

Testing

Extreme programming's approach is that if a little testing can eliminate a few flaws, a lot of testing can eliminate many more flaws.

- Unit tests determine whether a given feature works as intended. Programmers write as many automated tests as they can think of that might "break" the code; if all tests run successfully, then the coding is complete. Every piece of code that is written is tested before moving on to the next feature.
- Acceptance tests verify that the requirements as understood by the programmers satisfy the customer's actual requirements.

System-wide integration testing was encouraged, initially, as a daily end-of-day activity, for early detection of incompatible interfaces, to reconnect before the separate sections diverged widely from coherent functionality. However, system-wide integration testing has been reduced, to weekly, or less often, depending on the stability of the overall interfaces in the system.

Listening

Programmers must listen to what the customers need the system to do, what "business logic" is needed. They must understand these needs well enough to give the customer feedback about the technical aspects of how the problem might be solved, or cannot be solved. Communication between the customer and programmer is further addressed in the planning game.

Designing

From the point of view of simplicity, of course one could say that system development doesn't need more than coding, testing and listening. If those activities are performed well, the result should always be a system that works. In practice, this will not work. One can come a long way without designing but at a given time one will get stuck. The system becomes too complex and the dependencies within the system cease to be clear. One can avoid this by creating a design structure that organizes the logic in the system. Good design will avoid lots of dependencies within a system; this means that changing one part of the system will not affect other parts of the system.

Advantages

Robustness

Resilience

Cost savings

Lesser risks

Disadvantages

It assumes constant involvement of customers

Centered approach rather than design-centered approach

Lack of proper documentation

2.6 Managing Interactive Processes

Booch suggests that there are two levels of development:

- The macro process
- The micro process

Macro process

- Establish core requirements (conceptualization).
- Develop a model of the desired behavior (analysis).
- Create an architecture (design).
- Evolve the implementation (evolution).
- Manage post delivery evolution (maintenance).

Micro process

- Identify the classes and objects at a given level of abstraction.
- Identify the semantics of these classes and objects.
- Identify the relationships among these classes and objects.
- Specify the interface and then the implementation of these classes and objects

In principle, the micro process represents the daily activity of the individual developer, or of a small team of developers.

The macro process serves as the controlling framework of the micro process. It represents the activities of the entire development team on the scale of weeks to months at a time. The basic philosophy of the macro process is that of incremental development: the system as a

whole is built up step by step, each successive version consisting of the previous ones plus a number of new functions.

2.7 Basics of Software estimation

Estimation techniques are of utmost importance in software development life cycle, where the time required to complete a particular task is estimated before a project begins. Estimation is the process of finding an estimate, or approximation, which is a value that can be used for some purpose even if input data may be incomplete, uncertain, or unstable.

The four basic steps in software project estimation are:

- 1) Estimate the size of the development product. This generally ends up in either Lines of Code (LOC) or Function Points (FP), but there are other possible units of measure. A discussion of the pros & cons of each is discussed in some of the material referenced at the end of this report.
- 2) Estimate the effort in person-months or person-hours.
- 3) Estimate the schedule in calendar months.
- 4) Estimate the project cost in dollars (or local currency)

The major shortcomings of SLOC measure:

- No precise definition
- Difficult to estimate at start of a project
- Only a code measure
- Programmer-dependent
- Does not consider code complexity

2.8 Effort and Cost estimation techniques

The Effort and Cost estimation techniques are

- *Algorithmic models*, which use effort drivers representing characteristics of the target system and the implementation environment to predict effort.
- *Expert judgment*, based on the advice of knowledgeable staff.
- *Analogy*, where a similar, completed project is identified and its actual effort is used as the basis of the estimate.
- *Parkinson*, where the staff effort available to do a project becomes the estimate.
- *Price to win*, where the estimate is a figure that seems sufficiently low to win a contract.
- *Top-down*, where an overall estimate for the whole project is broken down into the effort required for component tasks.
- *Bottom-up*, where component tasks are identified and sized and these individual estimates are aggregated.

2.9 COSMIC full function points

- COSMIC FFP – Common Software Measurement International Consortium Full Function Point.
- COSMIC deals with decomposing the system architecture into a hierarchy of software layers.
- Unit is Cfsu(COSMIC functional size units).

A Data Movement moves one Data Group. A Data Group is a unique cohesive set of data (attributes) specifying an ‘object of interest’ (i.e. something that is ‘of interest’ to the user). Each Data Movement is counted as one CFP (COSMIC function point).

COSMIC recognizes 4 (types of) Data Movements:

- ☐ Entry moves data from outside into the process
- ☐ Exit moves data from the process to the outside world
- ☐ Read moves data from persistent storage to the process
- ☐ Write moves data from the process to persistent storage.

Function Points

Function points were defined in 1979 in *Measuring Application Development Productivity* by Allan Albrecht at IBM. The functional user requirements of the software are identified and each one is categorized into one of five types: outputs, inquiries, inputs, internal files, and external interfaces. Once the function is identified and categorized into a type, it is then assessed for complexity and assigned a number of function points. Each of these functional user requirements maps to an end-user business function, such as a data entry for an Input or a user query for an Inquiry. This distinction is important because it tends to make the functions measured in function points map easily into user-oriented requirements, but it also tends to hide internal functions (e.g. algorithms), which also require resources to implement.

There is currently no ISO recognized FSM Method that includes algorithmic complexity in the sizing result. Recently there have been different approaches proposed to deal with this perceived weakness, implemented in several commercial software products. The variations of the Albrecht-based IFPUG method designed to make up for this (and other weaknesses) include:

- Early and easy function points – Adjusts for problem and data complexity with two questions that yield a somewhat subjective complexity measurement; simplifies measurement by eliminating the need to count data elements.
- Engineering function points – Elements (variable names) and operators (e.g., arithmetic, equality/inequality, Boolean) are counted. This variation highlights computational function. The intent is similar to that of the operator/operand-based Halstead complexity measures.
- Bang measure – Defines a function metric based on twelve primitive (simple) counts that affect or show Bang, defined as "the measure of true function to be delivered as perceived by the user." Bang measure may be helpful in evaluating a software unit's value in terms of how much useful function it provides, although there is little evidence in the literature of such application.

The use of Bang measure could apply when re-engineering (either complete or piecewise) is being considered, as discussed in Maintenance of Operational Systems—An Overview.

- Feature points – Adds changes to improve applicability to systems with significant internal processing (e.g., operating systems, communications systems). This allows accounting for functions not readily perceivable by the user, but essential for proper operation.
- Weighted Micro Function Points – One of the newer models (2009) which adjusts function points using weights derived from program flow complexity, operand and operator vocabulary, object usage, and algorithm.

Benefits

The use of function points in favor of lines of code seek to address several additional issues:

- The risk of "inflation" of the created lines of code, and thus reducing the value of the measurement system, if developers are incentivized to be more productive. FP advocates refer to this as measuring the size of the solution instead of the size of the problem.
- Lines of Code (LOC) measures reward low level languages because more lines of code are needed to deliver a similar amount of functionality to a higher level language. C. Jones offers a method of correcting this in his work.
- LOC measures are not useful during early project phases where estimating the number of lines of code that will be delivered is challenging. However, Function Points can be derived from requirements and therefore are useful in methods such as estimation by proxy.

2.10 COCOMO II: A Parametric Productivity Model

COConstructive COSt MOdel II (COCOMO II) is a model that allows one to estimate the cost, effort, and schedule when planning a new software development activity. COCOMO II is the latest major extension to the original COCOMO (COCOMO 81) model published in 1981. It consists of three submodels, each one offering increased fidelity the further along one is in the project planning and design process. Listed in increasing fidelity, these submodels are called the

Applications Composition,

Early Design, and

Post-architecture models.

COCOMO II can be used for the following major decision situations

- Making investment or other financial decisions involving a software development effort
- Setting project budgets and schedules as a basis for planning and control
- Deciding on or negotiating tradeoffs among software cost, schedule, functionality, performance or quality factors
- Making software cost and schedule risk management decisions
- Deciding which parts of a software system to develop, reuse, lease, or purchase
- Making legacy software inventory decisions: what parts to modify, phase out, outsource, etc
- Setting mixed investment strategies to improve organization's software capability, via reuse, tools, process maturity, outsourcing, etc
- Deciding how to implement a process improvement strategy, such as that provided in the SEI CMM

The original COCOMO model was first published by Dr. Barry Boehm in 1981, and reflected the software development practices of the day. In the ensuing decade and a half, software development techniques changed dramatically. These changes included a move away from mainframe overnight batch processing to desktop-based real-time turnaround; a greatly increased emphasis on reusing existing software and building new systems using off-the-shelf software components; and spending as much effort to design and manage the software development process as was once spent creating the software product.

These changes and others began to make applying the original COCOMO model problematic. The solution to the problem was to reinvent the model for the 1990s. After several years and the combined efforts of USC-CSSE, ISR at UC Irvine, and the COCOMO II Project Affiliate Organizations, the result is COCOMO II, a revised cost estimation model reflecting the changes in professional software development practice that have come about since the 1970s. This new, improved COCOMO is now ready to assist professional software cost estimators for many years to come.

2.11 Staffing Pattern

Putnam was the first to study the problem of what should be a proper staffing pattern for software projects. He extended the classical work of Norden who had earlier investigated the

staffing pattern of general research and development type of projects. In order to appreciate the staffing pattern desirable for software projects, we must understand both Norden's and Putnam's results.

Norden's Work

- Norden studied the staffing patterns of several R&D projects.
- He found the staffing patterns of R&D projects to be very different from the manufacturing or sales type of work.
- Staffing pattern of R&D types of projects changes dynamically over time for efficient man power utilization.
- He concluded that staffing pattern for any R&D project can be approximated by the Rayleigh distribution curve.

Putnam's Work

- Putnam studied the problem of staffing of software projects.
- He found that staffing pattern for software development projects has characteristics very similar to R&D projects
- He adapted the Rayleigh-Norden curve to relate the no of delivered lines of code to the effort and the time required to develop the product.
- Initially less no of developers are needed.
- As the project progresses and more detailed work is performed, the number of developers increases and reaches a peak during product delivery
- After delivery, the no. of project staff falls consistently during product maintenance.

UNIT III –ACTIVITY PLANNING AND RISK MANAGEMENT

3.1 Objectives of Activity Planning

3.1.1 What is Planning

- Plans are nothing but planning is everything. Planning is a continuous process of refinement done during development.
- A detailed plan has to include the schedule of the project comprising of the start and the completion time of every activity defined. The actual achievement can be measured using the detailed plan.
- Planning process ensures that necessary resources needed at different stages are precisely available at requirement.
- Planning also produce a cash flow forecasting that indicates when the expenditure and he income takes place in the process.
- First of all, a plan must contain the start and completions of every activity that produces deliverables must be clearly visible in ensuring that the products of each activity are delivered on time.
- Every stage of the development plan must strive to achieve the objectives as the project moves from one to another.
- A plan must be defined with a set of targets that are achieved which can be measured. At the same instance, when target dates are not achievable the plan must be effectively modified to focus on the target.

3.1.2 Elements of Detailed Planned Activity

- Along with factors described with activity planning, the following elements play a very important role in achieving the target.
- The elements of a detailed planned activity are:
 - Feasibility assessment
 - Allocation of resources
 - Estimation of costs
 - Project coordination
 - Personal encouragement

Feasibility assessment

- Feasibility assessment talks about an very early stage describing whether it is feasible for the project to exist within the specified time constraint.
- A detailed plan will help in forecasting of the project as it progresses from one stage to other stages of activities.
- The feasibility factor also lies in the availability of resources that includes specialized staff to carry out the activities.

Allocation of resources

- The best way to allocate resources to the project depends on the availability factor.
- The project plan must analyze the available resources and the timescales for each and every activity.
- Additional usage of resources more than the stipulated timescale will result in slacking the progress of the project.

Estimation of costs

- The project plan must provide solutions to the following questions:
 - What is the total expenditure?
 - How much will the project costs?
 - What are the various estimating factors involved in the development process.
- These can be answered only when a detailed estimation of costs and timing is defined.

Project coordination

- Interaction and communication plays a vital role in handling complex projects.
- Effective team management must be established to carry out the activities in a well-coordinated manner.
- In particular, the availability of staff for a set of integrated project schedules must be carefully allocated with no period of idleness.

Personal encouragement

- Staff involved in the development process must be motivated in an effective way so that they achieve the target without any delay.

- The targets provided to the staff are monitored and personal encouragement must be given to individual staff if achieve the target on time.
- Activity planning helps in completing the project in minimum time with an nominal cost with the help of project schedules.
- To shorten the time limit, activities can be carried out in parallel based on the conditions defined for obtaining resources.
- Project scheduling activities includes the extension of timescale provided with constraints that can be relaxed to have effective usage.

3.2 Project Schedules

Stages of Project Schedules

- Every project must be developed with a plan showing the start and end of the activity along with the availability of resources for each activity.
- A project schedule is established based on the constraints defined for each activity.
- There are four stages involved in the creation of project schedule:
 - First stage of project schedule provides solutions to the questions like:
 - ❖ What are the activities have to be carried out?
 - ❖ What is the sequence of order in which each activity has to be handled?
 - The next stage involves the risk factors that affect each activity like:
 - ❖ Can risk occur in this activity?
 - ❖ How does a particular activity handle the risk?
 - ❖ What are the potential problems that can arise in risk handling?
 - Third stage of project schedule deals with allocation of resources:
 - ❖ How are resources allocated to specific activities?
 - ❖ What is the expected availability of resources?
 - ❖ What are the constraints defined for allocating a particular resource?
 - The final stage includes the schedule production:
 - ❖ What are the planned start and end dates for each activity?
 - ❖ What are the resources allocated to each activity?
 - ❖ Explain the detailed project schedule?

3.3 Activities

- Project and its activities must be clearly defined to achieve the target. An activity plan will contain the following factors:
 - A project is basically, composed of number of interrelated activities.
 - The initiation of a project happens only if atleast one activity is ready to start.
 - An activity is clearly defined with its start and end point that produce good deliverables.
 - Activity requiring resources must be analyzed well in advance and made available during the execution.
 - Some activities would depend on other activities for them to complete.
 - A project can attain its completion only when all activities have been completed.

3.3.1 Approaches to Identify Activities

- The various approaches used in identifying activities are:
 - Activity-based approach
 - Product-based approach
 - Hybrid approach

Activity-based approach

- In the activity-based approach, all the activities are listed and created for the project.
- This is achieved by a brainstorming session where the entire project team analysis the various activities needed at different stages with the help of similar projects.
- This approach usually generates the list of activities using a work breakdown structure (WBS).
- WBS helps in identifying the lowest level of effort i.e. the task required to complete a project by breaking down into lower sets of tasks.

Project

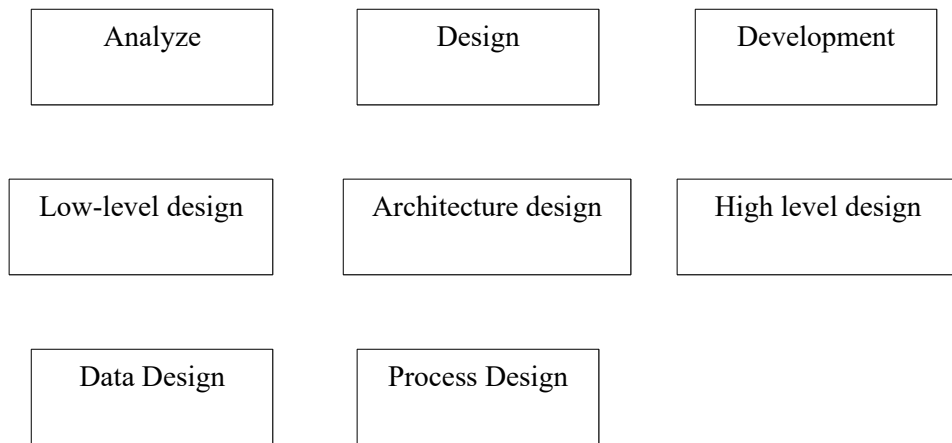


Figure - Activity-based approach Work Breakdown Structure

- Task defined at lower level includes everything that is required to complete the task at the higher level.
- The work breakdown structure provides an in-depth knowledge about the lowest level of activity that has to be completed.
- WBS is a refined structure that clearly defines the milestones that has to be achieved in accomplishing a specific task.
- The ordering of sequence of activities can also be done in this approach by defining those activities that have to be completed for others to start.
- In a purely activity-based approach, activities are identified and defined in five levels:
 - **Level 1** : Project – goals, objectives defined
 - **Level 2**: Deliverables – software, manuals, training
 - **Level 3** : Components – work items, modules, tests
 - **Level 4** : Work-packages – major work items, related tasks
 - **Level 5** : Tasks – responsibility of an individual in accomplishing it

Product-based approach

- The product-based approach produces a product breakdown structure along with a product flow diagram.
- The approach accepts the products as inputs which is transformed into an ordered list of activities.
- Product Flow Diagram do not leave out any activity from its ordered list and adopts a methodology which clearly specifies what are the products required and what are the activities required to produce the product.

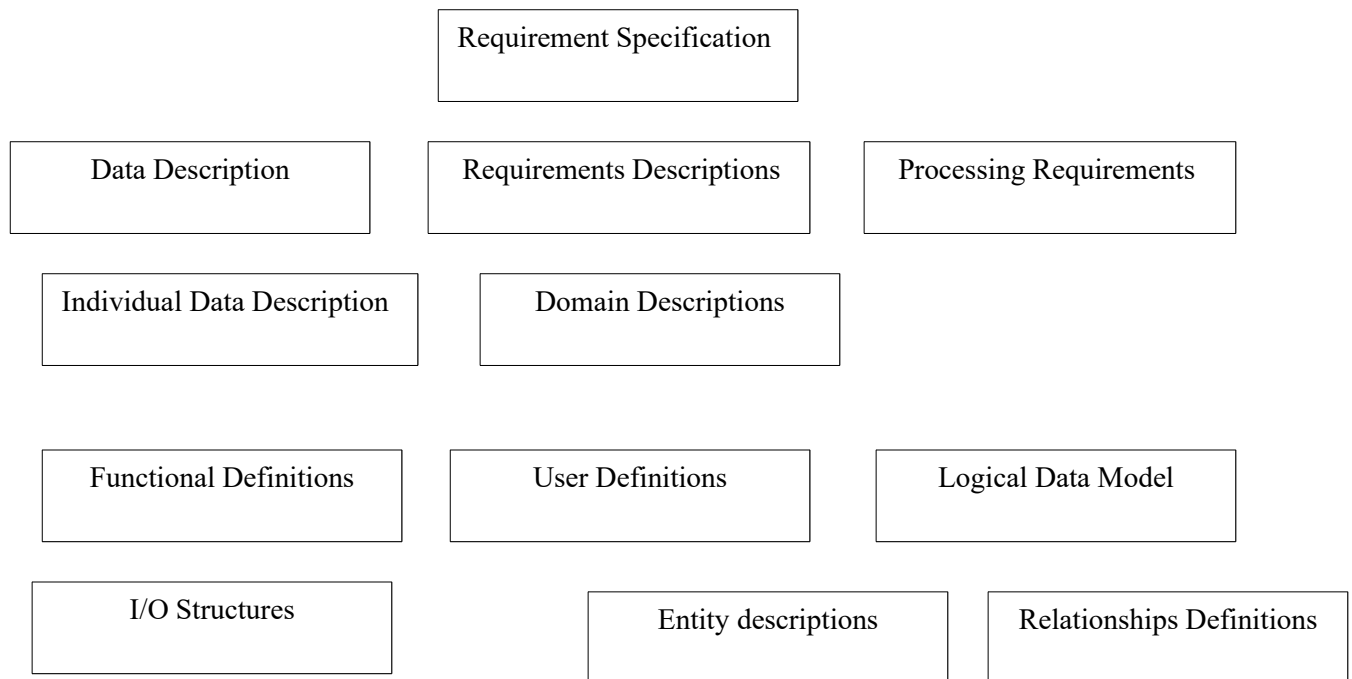


Figure - SSADM Product Breakdown Structure

- Using Structured Systems Analysis and Design Method (SSADM), a generic activity network can be derived for a project-specific product breakdown structure.
- The development of a PFD indicates the sequence of activities of the activity network.

Hybrid approach

- WBS deals with list of final deliverables whereas PBS deals in producing the products using the product flow diagram.

- Hybrid approach combines both the activity-based and product-based approach to structure both activities and products.
- Structuring of product-based or activity-based approach depend on the nature of the project type.

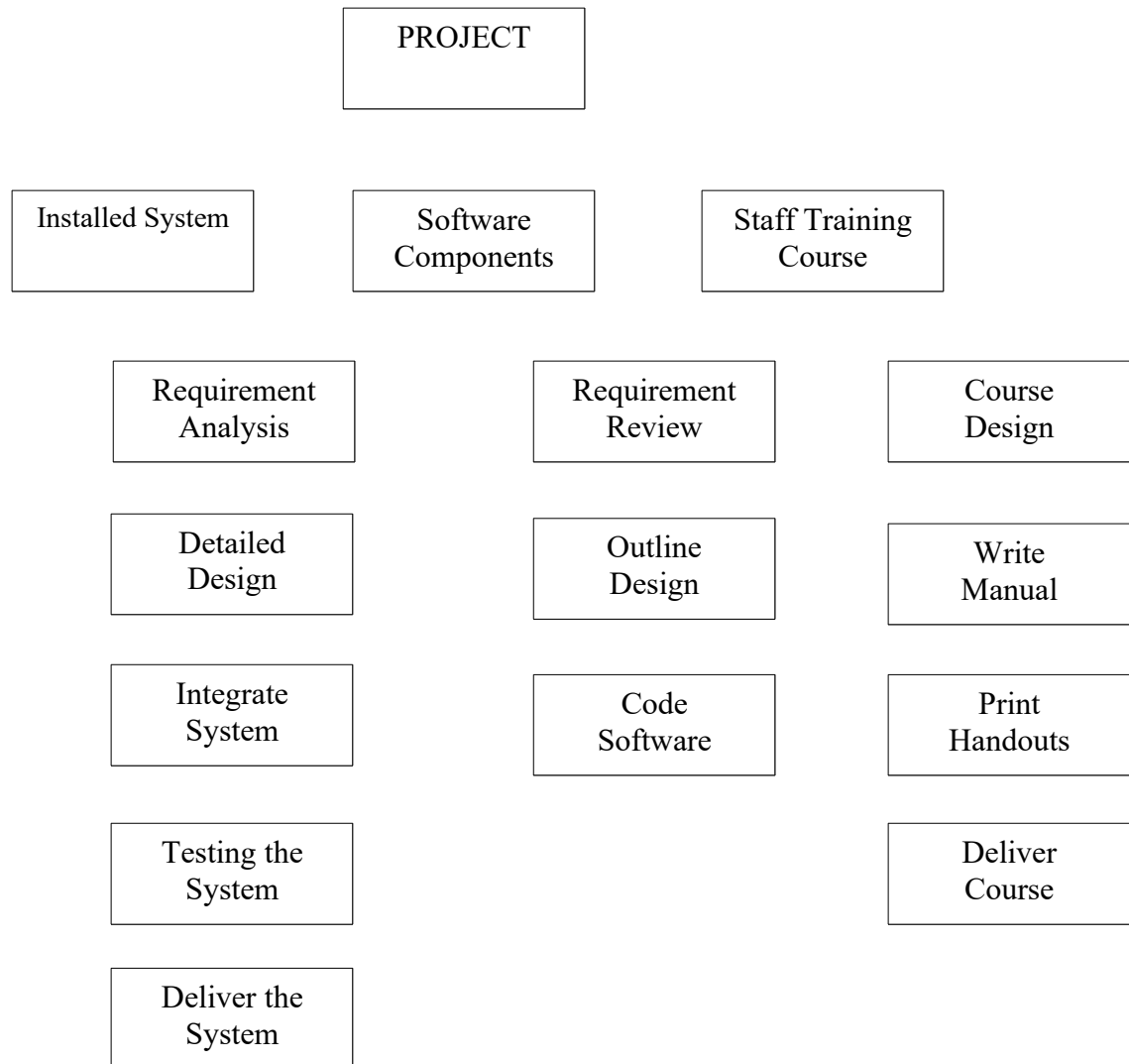


Figure - Hybrid Approach combining Activities and Products

3.4 Sequencing and Scheduling

- Scheduling is required for every activity that is planned along with the resources and can be represented using a bar chart.
- The chart describes the nature of the development process and the resources available for completing the specified activities.

Weeks	1	2	3	4	5	6	7	8	9	10	11	12
Person												
Requirements	■											
Design Module1		■										
Design Module2			■									
Design Module 3				■	■							
Code Module1			■	■	■	■						
Code Module2				■	■	■	■	■	■			
Code Module 3				■	■	■	■	■				
Integration									■			
System Acceptance										■	■	■

Figure - Bar chart representing Scheduling

- The chart defines two factors: sequencing of tasks and the schedule of the task. Scheduling includes the staff availability and the activities allocated to them.
- Combining sequencing – scheduling approach is suitable only for smaller projects and needs to be separated for complex projects as individual process.
- In case of larger projects, the logical relationship between the activities are grouped together and then scheduled for resources.

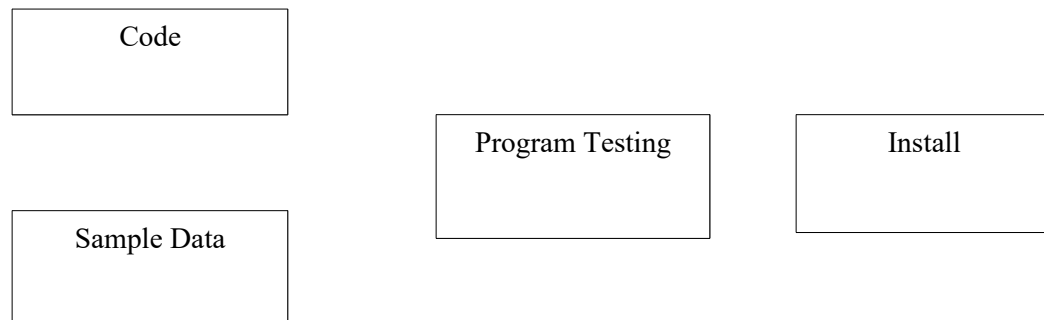
3.5 Network Planning Models

Framing a Network Model

- A network model can be formulated for the project scheduling techniques for the activities and their relationships as a graph.
- Most frequently used techniques are the Program Evaluation Review Technique (PERT) and the Critical Path Method.
- An activity-on-arrow (AOA) approach can be used to visualize the project as a network in which activities are shown as arrows joining the circles. Each node represents either the start or the end of an activity or a set of activities. This network can also be called as precedence network.
- An activity-on-node (AON) approach represents nodes as activities and the link between the nodes denote the precedence requirements.

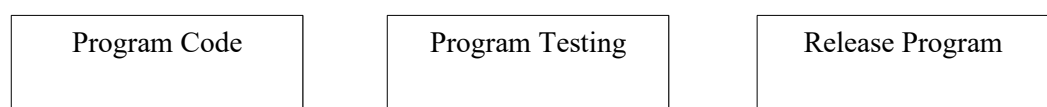
Constructing Precedence Networks

- There are some conventions used in the construction of precedence networks.
- **Only one start node and one end node must be defined for a project network:** There cannot be more than one start node for any project network and usually the duration of the start node is zero. Similarly, the completion of the project can be viewed by only one end node when the final activity is finished. If more than one start node or end node exists, it lead to confusion and uncertainty.
 - **Every node must have duration:** Any node that represents an activity must be provided with the duration for its execution. Here, the activities must be carried out in the sequenced order defined in the project schedule.
 - **Links do not have duration:** The relationship between activities are represented through links and generally does not have any duration for the establishment of creating it.
 - **Subsequent preceding activities are precedents:** Precedents are the successors of any preceding activity which are found out by the relationships defined.



Precedence Network

- **Flow of activities:** Activities are always started from the left most one and precedes in the forward direction. Usually, networks are drawn from left to right. Arrows can be drawn to show the flow of direction.
- **Loop free network:** Activity network must not contain any loop and if any loop exists, it results in error in the network. But certain process can be iterative in the development and such activities involved in iterative process should not be visually seen in the network. All network planning applications have the criteria of finding the loops and generate errors for both small and complex projects.



Correction of Errors

Diagnose Defects

Loop representing impossible sequence

- **Dangle-free network:** Dangling activities are never shown in the network. These leads to errors in subsequent analysis of the development process. For example, an activity named “Prepare User Manual” should not be defined in the network as a non-connectivity activity, instead must be defined before the installation of the software.

Program
Design

Program
Code

Program
Testing

Program
Installation

Preparation of
User manual

Activity network representing a Dangle

Program
Design

Program
Code

Program
Testing

Program
Installation

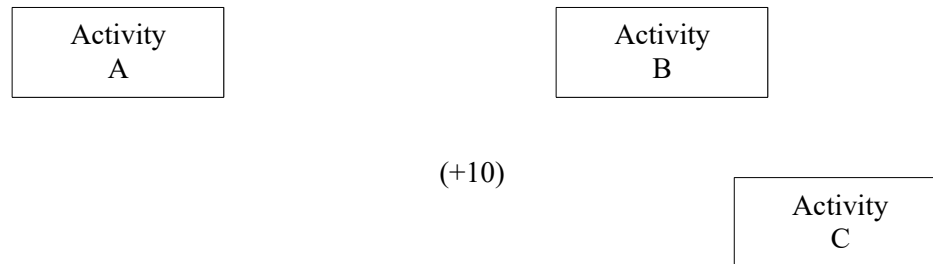
Program
Sign - off

Preparation of
User manual

Modified Activity network with Dangle-free

Lagged and Hammock activities

- The start or finish of one or more activity is slipped-out (lagged) or pulled-up(led) by a certain period.
- A time lag is represented for activities that occur in parallel when time exceeds and the duration is shown on the linking arrow.
- Usually, a lag is indicated with a positive number and a lead uses a negative number added to the start or finish time of their connected activities.



Lag and Lead Activities

- Here, activity B cannot start until activity A is finished. At the same time, activity C cannot be started until 10 days after activity A is finished.
- Hammock activities have zero duration and assumed to start at the same time for each activity and end at the same time for the last one. These activities normally represent overhead costs and resources that occur at regular intervals over the set of activities.

3.6 Forward Pass & Backward Pass Techniques

- The logical network model represents the inter-relationships between the activities and is used in estimating the duration of the activity.
- Critical Path Method ensures that the planned project must be completed as quickly as possible. It also governs those activities that have delay in execution which can affect the overall project schedule.

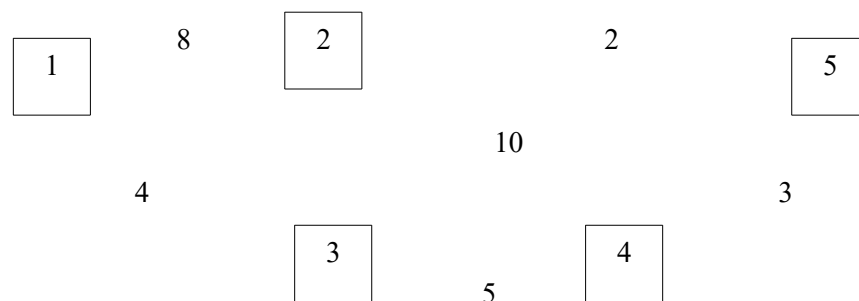
- The critical path method analyses the precedence of activities to predict the total project duration.
- The focus is based on the slack, free float and path float available between the activities. The method calculates which sequence of activities has the least amount of schedule flexibility.
- CPM analysis starts with a WBS that has single point estimates for each activity and uses the precedence diagramming method to relate the precedence in the network.
- With the network drawn, two-pass analysis can be performed through the network of activities and calculate the node quantities for each activity.

Forward Pass

- Forward pass is used to calculate the earliest dates on which each activity may be started and completed.
- The steps involved in forward pass are:
 - Start at the start node
 - Compute the top pair of numbers
 - Always add the duration to the connecting node's earliest finish time.
- For example, given the following data,

Activity :	1-2	1-3	2-4	2-5	3-4	4-5
Duration :	8	4	10	2	5	3

The network is drawn below:

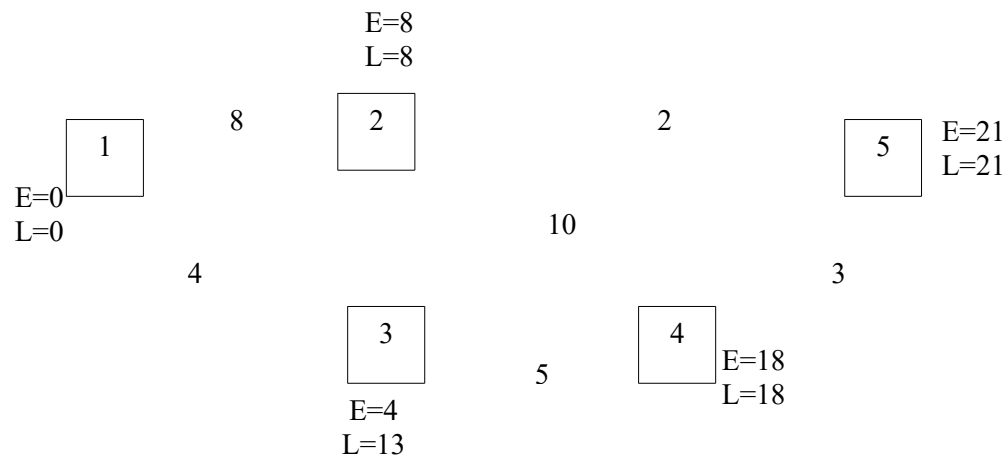


Backward Pass

- Backward pass is used to calculate the latest dates at which each activity may be started and completed without delaying the end date of the project..
- The steps involved in backward pass are:
 - Start at the end node
 - Compute the bottom pair of numbers

- Always subtract the duration from the connecting node's earliest start time.
- Considering the above example and representing network diagram with both the passes:

The network with the earliest and latest occurrence of events is drawn below:



Network diagram with earliest and latest time

3.7 Critical Path Method

- The critical path is a single path that defines the duration of the project.
- Activity float is a measure which calculates the difference between the activity's earliest start date and the latest start date.
- An activity with a float value to be zero is called critical because delay in carrying out the activity will affect the project completion date.
- Free float is the delay time taken by single activities that do not affect other activities where as interfering float represents how much the activity can be delayed without affecting the end date.
- Atleast one path exists in the network joining the critical activities which forms the critical path of the network.
- Critical path must be established because monitoring critical activities have a greater impact on the completion of the project and it shortens the overall duration of the project.

➤ For the same example, the critical path and the activity float is calculated as:

Activity	Duration Days	Earliest		Latest		Activity Float / Total Float
		Start ES	Finish EF EF = ES + t_{ij}	Start LS LS = LF - t_{ij}	Finish LF	
1 – 2	8	0	8	0	8	0
1 - 3	4	0	4	9	13	9
2 - 4	10	8	18	8	18	0
2 - 5	2	8	10	19	21	11
3 – 4	5	4	9	13	18	9
4 - 5	3	18	21	18	21	0

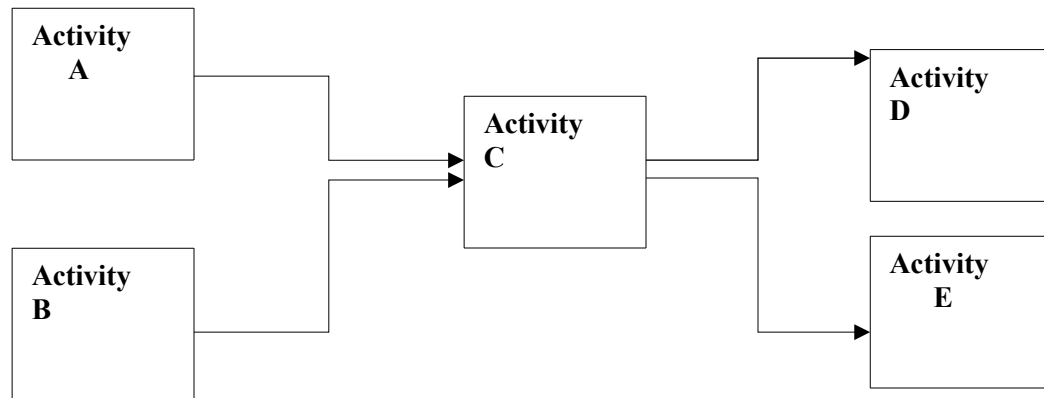
The critical path is 1→2 →4 →5 and all the activities in the critical path are termed as critical activities.

Rules and Conventions

- The following rules apply for the activity–on–arrow network. They are
- The project must have only one start node and only one end node.
 - The link between nodes represents the duration but the nodes do not have any duration.
 - The nodes are numbered sequentially and the arrows move only from left to right.
 - The activity-on-arrow network should not contain any loop or dangle.

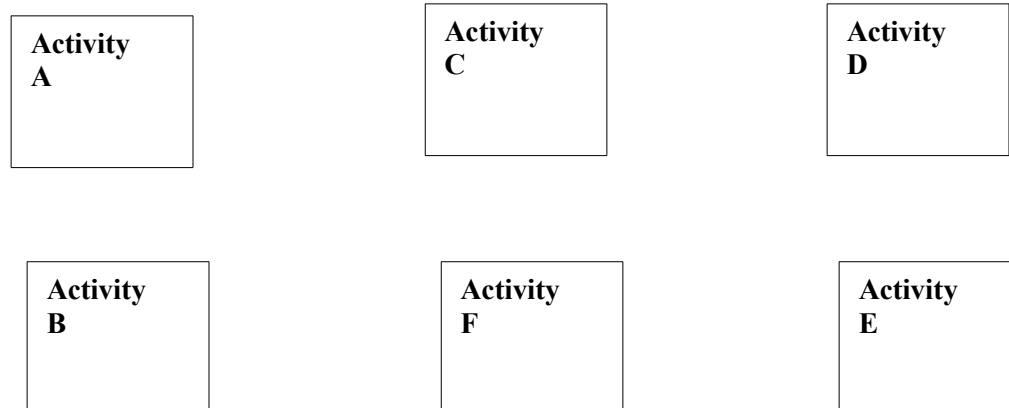
Dummy Activities

- When there are two different paths having the common activity then, it can be represented using dummy activity.
- The model below describes a situation where the third node is a common event.



Common activity in two paths

- The problem can be solved by introducing a dummy activity for the third node as shown below where activity F is included as an dummy activity.



Dummy activity supporting the common paths

- The paths are separated and made as independent paths by dotted lines in the network diagram.
- Dummy activities have zero duration and do not use any resources.
- These problems can be overcome by representing the network with activity-on-node.
- An activity-on-node is illustrated below:

Lagged Activities and Labelling Activities

- Lagged activities can be represented using ladder technique when activities are done in parallel.

- The start or finish of one or more activity is slipped-out (lagged) or pulled-up(led) by a certain period.
- A time lag is represented for activities that occur in parallel when time exceeds and the duration is shown on the linking arrow.
- The lagged activity must be completed before the subsequent activity is started.
- There are many conventions used in labeling the activity. Information present on the network diagram describes about the specific activity and the related details are separately maintained in a different table.

Network Analysis

- Network analysis is similar to activity-on-node and can be done in two passes.
- **Forward pass** calculates the earliest dates of events and activities. For events, are recorded on the network diagram and activities in the activity table.
- **Backward pass** calculates the latest date for an event is the latest start date for all activities that arise from the event. When there are multiple activities happening, the earliest of the latest start dates for those activities are considered.
- **The critical path** is the longest path through the network but slack is used in identifying the path.
- Slack measures the difference between the earliest date and the latest date for an event without affecting the end date of the project. The nodes that form the critical path will have a zero slack.

3.8 Framework in Risk Management

- Risks are uncertain events that happen in the development process. Basically, risks have to be identified, analyzed and prevented from occurring in the project.
- The framework for dealing with risk are
 - Risk identification
 - Risk assessment
 - Risk planning
 - Risk monitoring

3.8.1 Risk Identification

- Identification of risks involves the following techniques namely,
 - Brainstorming
 - Checklists
 - Casual mapping
- All stakeholders identified in the project are brought together to have a brainstorming session before the project commences.
- These stakeholders identify the various features of the project and analyze the problems that can arise in due course.
- The outcome of the discussions of these sessions is beneficial in view point of the development process because almost all kinds of risks that the project will face are analyzed.
- Checklists are those risks that occur frequently in software development projects. These risks contain a list of specialized software development risks that occur.
- Every stakeholder must undergo a through checking of this list and find out the kind of risk that can happen in their projects.
- Project managers will have a separate list of risks solely associated with the software process. Any new kind of risks that happens in any of the projects can be added on to the organizational risk list.
- Casual mapping can be represented as a cognitive map that describes the causes and effects that influence the outcomes in the activity development. The outcome can be a negative or a positive influence depends on the particular factor. For example, high staff turnover can be a positive factor but unstable requirements have a negative impact.
- Casual maps usually represent people's perspective towards the development of the project.

3.8.2 Risk Assessment

- Most frequent risks that occur causes more damage to the process. Risk exposure for every risk has to be calculated with the probability of occurrence.
- Risk exposure is defined by

$$\text{Risk exposure} = (\text{value of damage}) \times (\text{probability of occurrence})$$

- Here the potential damage is assessed the money value of the development process.
- Few risk exposure assessments are listed below:
 - Requirement specification changes during coding phase
 - Staffs inability to complete the task assigned affecting the critical activities.
 - Specification process takes much longer than expected.
 - Module testing produces errors of design phase.
- Managers try to produce very precise estimates of loss or they expect something to happen. It is the duty of the managers to prioritize the risk and handle them giving due importance to its existence.
- The potential damage and the likelihood of risk are described by qualitative descriptors, depicts the causes and the impact of the project are shown below:

Probability level / Impact level	Risk Probability (measured in chance of happening)	Cost (above budgeted expenditure)
High	Greater than 50%	More than 30%
Significant	30 – 50%	20 – 29%
Moderate	10 – 29%	10 – 19%
Low	Less than 10%	Within 10%

Qualitative descriptors of Risk Probability and Cost with range values

A probability impact grid or summary risk profiles are described in a matrix which indicates the position of risk. The top right of the matrix denotes the tolerance line with serious risks levels.

3.8.3 Risk Planning

- Risk planning involves the following factors:
 - **Risk acceptance:** A risk that has already occurred according to he prioritization process cannot be avoided. Accept the risk that happens and minimizes the damage and the costs of action.
 - **Risk avoidance:** Some risks that happen regularly can be categorized and avoided before it occurs.

- **Risk reduction:** Precautionary measures are taken to reduce the probability of risk. Risk reduction attempts to reduce the occurrence of risk whereas risk mitigation ensures that the risk impact is much lesser when actually occurs.
- **Risk transfer:** Certain complex risks can be transferred to other organizations where experienced professionals can carry out the possibility of its occurrence.

3.8.4 Risk Monitoring

- Risk monitoring is a planned process of assessing whether the predicted risks occur or not. It also collects information of the future risk analysis and attempts to determine what has caused the particular risk.
- Project manager monitors the following factors:
 - General attitude of team members

RISK REGISTER RECORD				
Risk ID	Risk title			
Owner	Date raised		Status	
Description of Risk				
Description of Impact of risk				
Recommended risk mitigation				
Probability / Impact values				
			Impact	
	Probability	Cost	Duration	Quality
Pre-mitigation				
Post-mitigation				
History of action				
Date	Action		Actor	Outcome

Risk Register Page

3.9 PERT

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 methodology developed by the U.S. Navy in the 1950s to manage the Polaris submarine missile program.

PERT uses time as a variable which represents the planned resource application along with performance specification. In this technique, first of all, the project is divided into activities and events. After that proper sequence is ascertained, and a network is constructed. After that

time needed in each activity is calculated and the critical path (longest path connecting all the events) is determined.

- PERT is a project management technique, whereby planning, scheduling, organising, coordinating and controlling of uncertain activities is done. CPM is a statistical technique of project management in which planning, scheduling, organizing, coordination and control of well-defined activities takes place.
- PERT is a technique of planning and control of time. Unlike CPM, which is a method to control costs and time.
- While PERT is evolved as research and development project, CPM evolved as construction project.
- PERT is set according to events while CPM is aligned towards activities.
- A deterministic model is used in CPM. Conversely, PERT uses probabilistic model.
- There are three times estimates in PERT i.e. optimistic time (t_o), most likely time (t_m) pessimistic time (t_p). On the other hand, there is only one estimate in CPM.
- PERT technique is best suited for a high precision time estimate, whereas CPM is appropriate for a reasonable time estimate.
- PERT deals with unpredictable activities, but CPM deals with predictable activities.
- PERT is used where the nature of the job is non-repetitive. In contrast to, CPM involves the job of repetitive nature.
- There is a demarcation between critical and non-critical activities in CPM, which is not in the case of PERT.
- PERT is best for research and development projects, but CPM is for non-research projects like construction projects.
- Crashing is a compression technique applied to CPM, to shorten the project duration, along with least additional cost. The crashing concept is not applicable to PERT.

Three estimates are produced for each activity

- *Most likely time (m)*
- *Optimistic time (a)*
- *Pessimistic (b)*

$$\text{Expected time'} te = (a + 4m + b) / 6$$

$$\text{Activity standard deviation'} S = (b-a)/6$$

Expected time: Helps to carry out a forward pass through a network similar to CPM

Activity standard deviation: Used as ranking measure of the degree of uncertainty or risk for each activity

Activity	Optimistic (a)	Most likely (m)	Pessimistic (b)	Expected te	Standard deviation s
A	5	6	8	6.17	0.5
B	3	4	5	4.00	0.33
C	2	3	3	2.83	0.17
D	3.5	4	5	4.08	0.25
E	1	3	4	2.83	0.5
F	8	10	15	10.50	1.17
G	2	3	4	3.00	0.33
H	2	2	2.5	2.08	0.08

3.10 Monte Carlo Simulation

A Monte Carlo method is a technique that involves using random numbers and probability to solve problems. This method is often used when the model is complex, non linear or involves more than just a couple uncertain parameters. A simulation can typically involve over 10,000 evaluations of the model, a task which in the past was only practical using super computers. The Monte Carlo method is just one of many methods for analyzing uncertainty propagation, where the goal is to determine how random variation, lack of knowledge, or error affects the sensitivity, performance, or reliability of the system that is being modeled.

Monte Carlo simulation is categorized as a sampling method because the inputs are randomly generated from probability distributions to simulate the process of sampling from an actual population. So, we try to choose a distribution for the inputs that most closely matches data we already have, or best represents our current state of knowledge. The data generated from the simulation can be represented as probability distributions (or histograms) or converted to error bars, reliability predictions, tolerance zones, and confidence intervals.

The main steps involved in carrying out Monte Carlo Simulation for a project consisting of n activities are as follows.

Step 1: Express the project completion time in terms of the duration of the n activities (x_i , $i=1$ to n) and their dependences as a precedence graph, $d = f(x_1, x_2, \dots, x_n)$.

Step 2: Generate a set of random inputs, $x_{i1}, x_{i2}, \dots, x_{in}$ using specified probability distributions.

Step 3: Evaluate the project completion time expression and store the results in d_i .

Step 4: Repeat steps 2 and 3 for the specified number of times.

Step 5: Analyze the results d_i using histograms, summary statistics, confidence intervals, etc.

3.11 Resource Allocation

Resource allocation is the assignment of available resources to various uses. In the context of an entire economy, resources can be allocated by various means, such as markets or central planning. In project management, resource allocation or resource management is the scheduling of activities and the resources required by those activities while taking into consideration both the resource availability and the project time.

The resource allocation will normally be a number of schedules, including

- Activity Schedule – including the planned start and completion dates for each activity
- Resource Schedule – showing the dates on which each resource will be required and the level of that requirement
- Cost Schedule – planned cumulative expenditure incurred by the use of resources over time

In general, resources will fall into one of seven categories:

- Labour – Members of the project team
- Equipment – Workstations and other communicating and office equipments
- Material – Items that are consumed
- Space – Office space
- Services – Some specialist services telecommunicating
- Time – Offset against the other primary resource

Identifying Resource Requirements

- What resources are required along with the expected level of demand
- Consider each activity
- Identify required resources

Scheduling Resources

- Allocating resources for one activity limits flexibility for resource allocation and scheduling of other activities

Priorities resource allocation

- Total float priority

Activities are ordered according to their total float. Those with the smallest float are assigned the highest priority

Ordered list priority

⌘ Ordered according to predefined criteria

- } Shortest critical path – Critical activities
- } Shortest non-critical activity
- } Non-critical activity with least float
- } Non-critical activities

3.12 Creation of Critical Patterns

Scheduling resources can create new critical paths. Delaying the start of an activity because of lack of resources will cause that activity become critical if this uses up its float.

Manage the allocation of resources within programmers

The resources of an organization consist of people, materials, equipment, knowledge and time. Organizations typically have limited resources; therefore, tradeoffs on what project resources are expended and when are made every day within organizations. A resource allocation plan is an important tool in effective management of scarce resources. The timing of the need of those resources can be and should be determined within the project schedules. A resource plan, which describes the type of resource needed and the timing of that need, is critical to effective resource management. As the project schedule changes, the resource plan must also be flexible enough to adjust as these changes occur.

Examples

Allocating resources is fairly self-explanatory. If allocating stone for building a house, the project manager must ensure that she procures enough stone to complete the project. Regarding leveling, if renting equipment, the project manager must ensure it will be used

steadily rather than sporadically rented and returned. If contracting carpenters, the project manager should aim to strive to keep a set number of carpenters working at a set number of hours for the duration of the project to ensure consistency. Carpenters may have difficulty scheduling more sporadic hours into their schedule, meaning the firm might then have to contract more workers, leading to inconsistent results. Meanwhile, materials don't necessarily need to be leveled as they have been purchased rather than rented or paid by the hour.

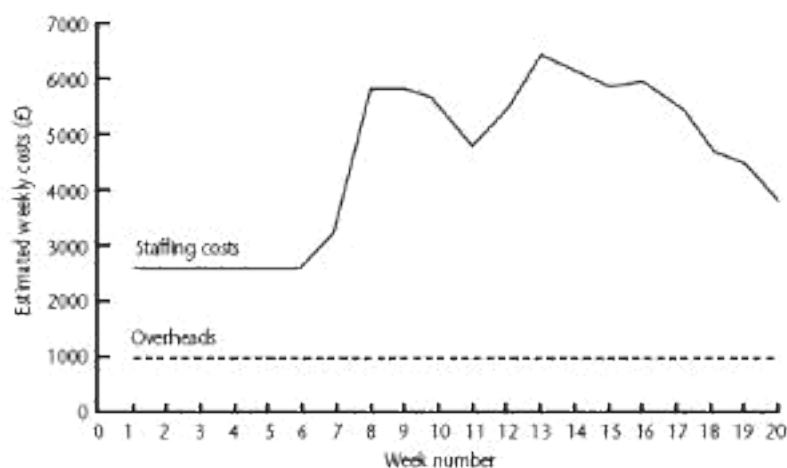
3.13 Cost Schedules

Calculating cost is straightforward where organization has standard cost figures for staff and other resources. Staff costs includes not just salary, but also social security contributions by the employer, holiday pay etc. Timesheets are often used to record actual hours spent on each project by an individual. One issue can be how time when a staff member is allocated and available to the project, but is not actually working on the project, is dealt with. Overheads e.g. space rental, service charges etc. Some overheads might be directly attributable to the project, in other cases a percentage of departmental overheads may be allocated to project costs. Usage charges are some charges can be on a 'pay as you go' basis e.g. telephone charges, postage, car mileage – at the planning stage an estimate of these may have to be made.

In general, costs are categorized as follows.

- Staff Costs
- Overheads
- Usage Charges

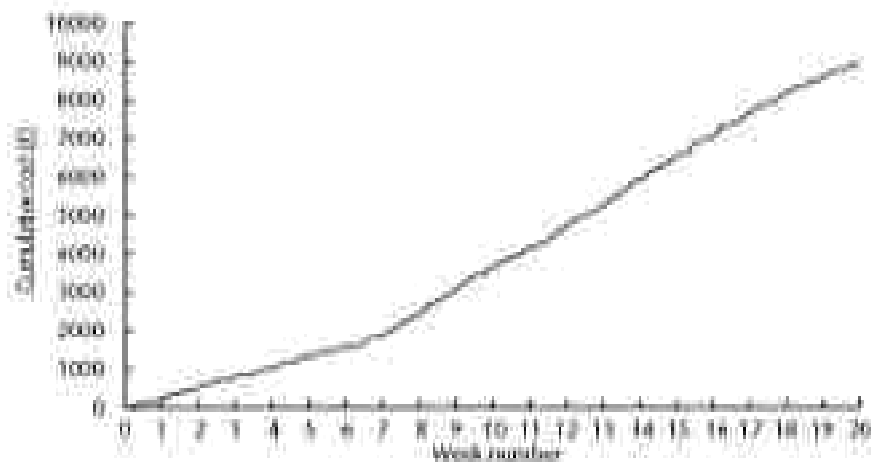
Cost profile



This shows how much is going to be spent in each week. This could be important where an organization allocates project budgets by financial year or quarter and the project straddles more than one of these financial periods

Accumulative costs

The project manager will also be concerned about planned accumulative costs. This chart can be compared to the actual accumulative costs when controlling the project to assess whether the project is likely to meet its cost targets.



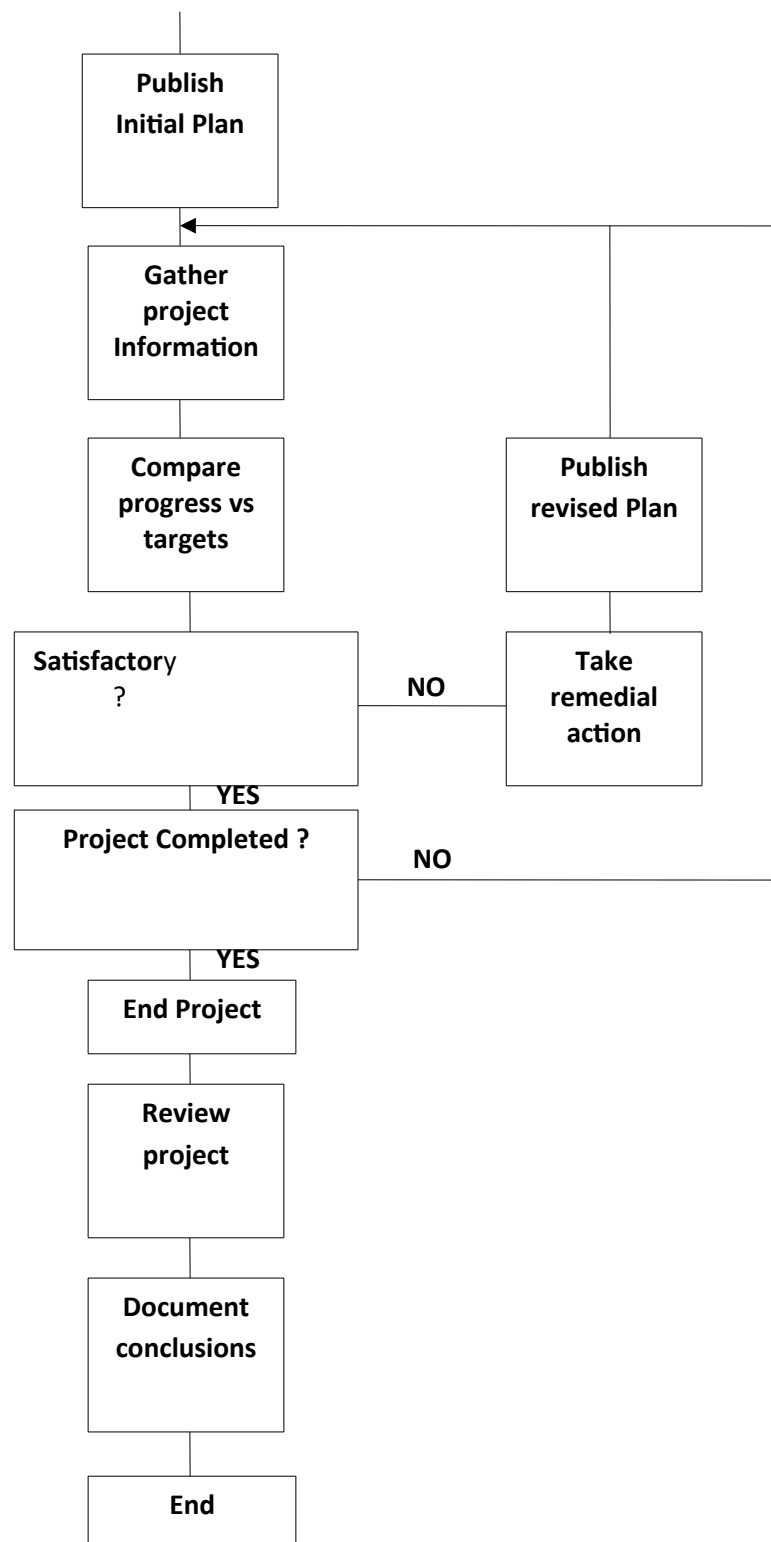
UNIT IV - PROJECT MANAGEMENT AND CONTROL

4.1 Framework of Management and Control

4.1.1 Creating Framework

- After the project starts its execution, the project must be carefully monitored to ensure the project's progress.
- Monitoring process focuses on comparing the actual output with the expected one and reviews the schedule to fit on target.
- Regular monitoring of the project is needed to have more control over the project. Always the expected outcomes are compared with the actual ones and analyzed whether there is any slack in the planned process.
- Project control is a continuous process of monitoring the progress of the project plan and it also includes re-planning of activities if necessary.
- The experience gained from the current project can be taken as an input over future project establishment of activities.
- Generally, revising the planning strategy is due to:
 - Delay in completion of the project within the target time
 - Quality factors
 - Inadequate functionality in adopting newer techniques
 - Actual estimation is above the estimated one.
- A typical project control cycle is depicted below:

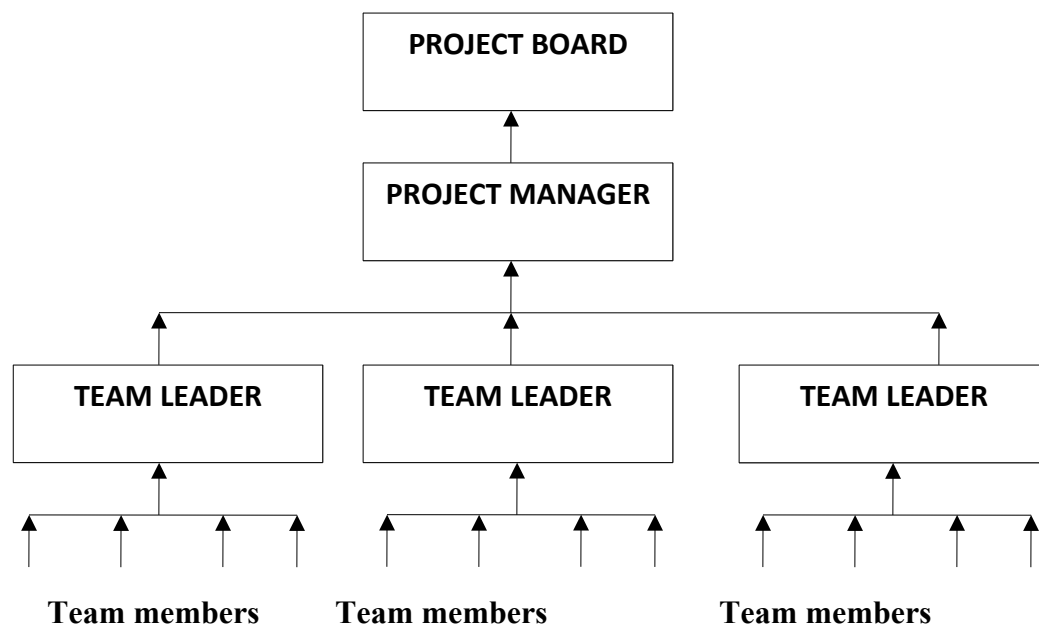
Start



Project Control Cycle

4.1.2 Project Reporting Structures

- Project steering committee or the Project Board has the overall responsibility of the project's progress in achieving the target.
- Project manager has the day-to-day responsibility of governing the development of the project. These managers assign individual responsibilities to different teams under a team leader.



Project Reporting Structures

- The diagram represents a structure for a medium-sized project where team leaders can directly report to the project managers.
- Every team consists of a group of team members assigned with specific tasks. These members represent the respective team according to the work allocated.
- Team leaders organize and collect team related information and report to the project manager.
- The project manager in-turn generates a project-level report of the progress of the project and report to the steering committee.

4.1.3 Categories of Reporting

- Reporting is broadly classified as formal and informal reporting. The basic types of reports associated with formal and informal reporting includes regular and ad hoc types.
- **Formal regular** types can be oral or written. The standard oral communication of minutes are kept whereas written type gets the reporting issues in a separate written format.
- **Formal ad hoc** are mostly received information of different levels towards the end of the project and generate written reports.
- **Informal, oral and ad hoc** provides early warning to the system and must be backed up by formal reporting procedures.

4.1.4 Progress Assessment

- Based on the information collected from various levels at regular intervals during the development of the project measures the progress assessment.
- The information can however, measure the project's objectives in determining whether the project can produce deliverables or not.
- Every single activity will not yield a deliverable work product but a group of activities can achieve the specified tangible product.
- Usually, the assessment process is carried out by the team members who are associated with the project activities.
- **Checkpoints** can be used to check the initial activity plan which may govern specific events in generating report or a deliverable.
- Team leaders will have to assess the project daily while the project leaders can do it on a weekly basis. Higher level members generate less reporting than their subordinates.
- **Review points or control points** can be set at different points in the project life cycle to review the progress of the project.

4.2 Collection of data project termination

- Collecting information of the project progress at regular instances provides much control over the project.
- However, gathering of partial completion of activities can be used to calculate the remaining work needed to complete.

- Intermediate products that are achieved can specify a milestone in the development of the project.

4.2.1 Partial Completion Reporting

- The staff time related to a specific project indicates the work that has to be carried out by the particular staff.
- Every organization uses an accounting method to calculate the charges of their employees. However, the information related to project schedule is not shown in this report.
- Timesheets can be maintained on a weekly basis to measure the staff involvement in the development process.

WEEKLY TIME SHEET OF INDIVIDUAL TEAM MEMBER						
Staff : _____			Week Ending : _____			
Worked Hours						
Project	Activity Code	Description	Hours this week	Percentage of Completion	Scheduled Completion	Estimated Completion
Total Worked Hours :						
Non-worked hours						
Code	Description		Hours this week	Comment & Authorization		
Total Non- Worked Hours :						

Weekly Timesheet

- Weekly timesheets contain the breakdown activities and holds information about the scheduled and estimated completion time of individuals and do not contain the project completion dates.

4.2.2 Reporting Risk

- Risk reporting uses a traffic light method concept and consists of the following steps:
 - Identify the first level elements for assessment
 - Break the first level elements into second level elements
 - Assess the second level elements and mark the color as
 - Green – on target**
 - Amber – not on target but recoverable**
 - Red – not on target and difficult to recover**
 - Review all the second level elements to reach the first level assessments
 - Review both first and the second level assessments to produce an overall assessment
- This method only focuses on non-achievement factors and do not mention about any delay in the development process.
- Assessment forms can be used to evaluate the overall status of the project.
- Critical activities denoted by red color must be reconsidered during the revision of project schedule.

4.3 Visualizing Progress

- Collected data cannot be represented as arrived. It has to be shown visually so that everybody involved in the project work is pleased about its progress. Presenting effectively plays a vital role in the future of the project

4.3.1 Categories of Visualizing Progress

- The techniques that are used in visualizing project progress are:
 - Gantt chart
 - Slip Chart
 - Ball Chart

- These are explained in detail in the following sections.

4.3.2 Gantt Chart Technique

- Gantt chart is the most simple and the oldest form of representing the progress of the project.
- It consists of an activity bar that indicate the scheduled activity dates and the duration along with the activity floats.

Planned Week	15					16					17					18				
Work Days	M	T	W	T	F	M	T	W	T	F	M	T	W	T	F	M	T	W	T	F
Code & Test Module 1																				
Code & Test Module 2																				
Code & Test Module 3																				
Code & Test Module 4																				
Specify Overall System																				
Check Specifications																				
Review Meetings																				

- The progress reports of the activity are normally represented as a shaded activity bar which indicates the percentage of activity completion.

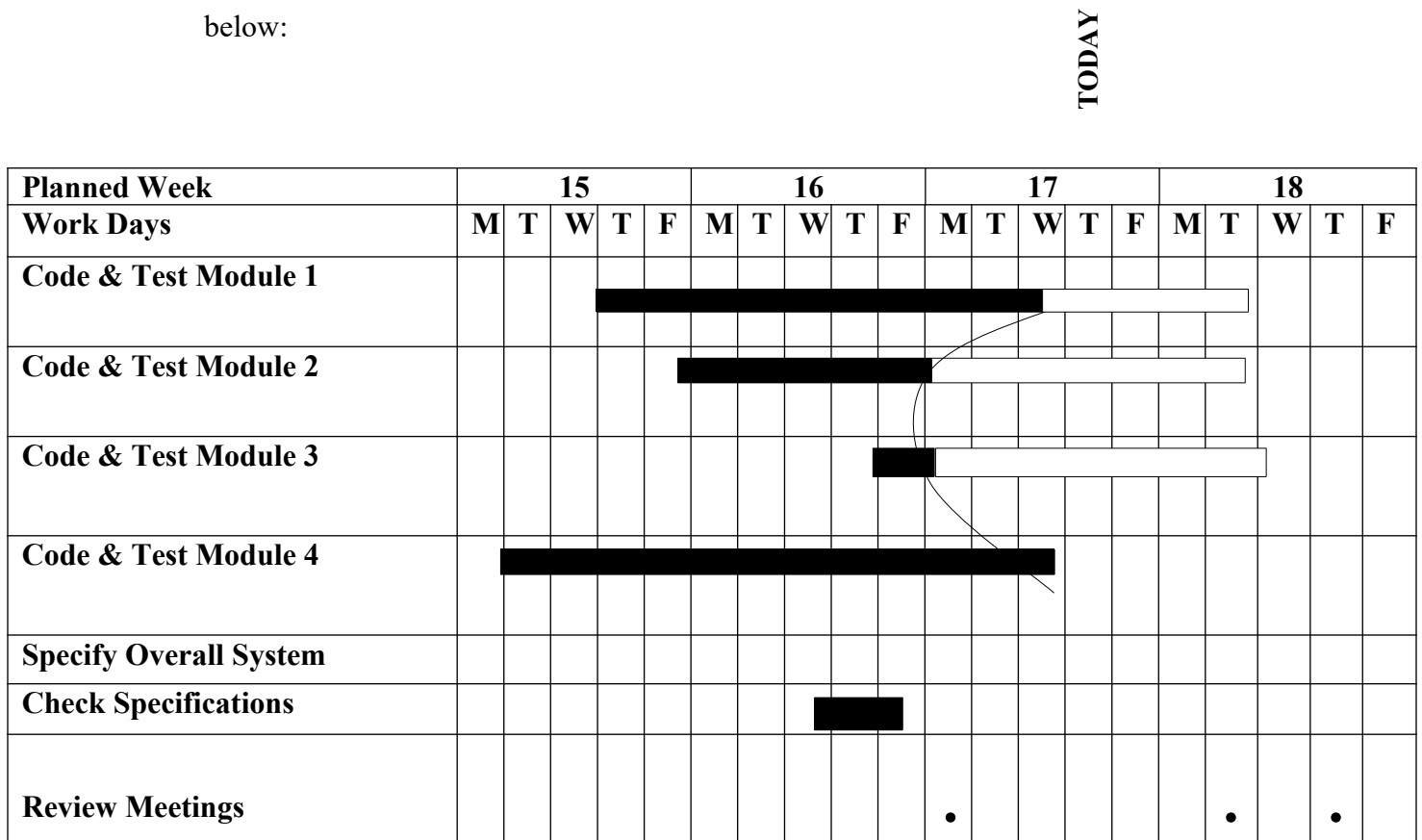
TODAY

Sample Gantt Chart

- For example in the figure, the code and test module activity of X is ahead of the completion process whereas the third activity Z is lacking behind in its schedule

4.3.3 Slip Chart Technique

- A Slip chart provides an alternative view of Gantt chart by providing a visual indication of those activities which are not on schedule.
- The chart indicates that, the more there is a bend in the line the greater the variation in the project plan.
- If the slip line deviates more towards the non-achievement of project objectives, then it has to be reconsidered.
- The same figure used to represent Gantt chart is modified to Slip chart and depicted below:

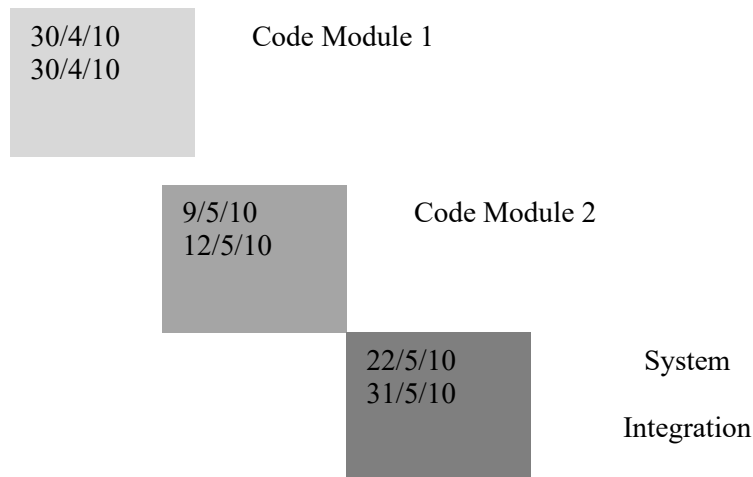


Slip Chart

- Additional slip lines can be included at regular intervals as they are built up which provides the project manager a clear idea about the project's progress.

4.3.4 Ball Chart Technique

- Ball charts are represented in the form of circles that indicate the start and the end point completion of activities.
- Initially, the circles contain the original scheduled dates and when revisions are done, the second dates are introduced inside the circle until the activity is started or completed.
- Circles of bar chart will at most contain only two dates the original and the revised one or the original and the actual dates.
- Ball charts are pictorially shown as below:



Ball Chart

- An activity is denoted by a red circle (colored darker in the figure) when the start and the end dates are later than the target dates whereas green circle (colored lighter in the figure) denotes that the activity is ahead of its schedule.
- The color to the circles reminds the project team about the status of each activity.
- In general, all the three types of chart techniques do not show clearly the slippage in the project completion date for the project life cycle. This is overcome by timeline charts.

4.3.5 Timeline Charts

- Timeline usually records and displays the target changes during the project life cycle.
- The chart represents the planned time along the horizontal axis and the actual time along the vertical axis.

- A line down the horizontal axis represents the scheduled activity completion dates and the slip in the line indicates a delay in the respective activities.
- This timeline chart is used to calculate the duration of execution of the project as a part of post-implementation review.

Planned Time / Week Numbers

Actual Time / Week Number

	1	2	3	4	5	6	7	8	9
	MTWTF	MTWTF	MTWTF	MTWTF	MTWTF	MTWTF	MTWTF	MTWTF	MTWTF
1			ing system		User requirements	in offline layout			
2								Draft tender	Issue Tender
3									
4									
5									
6									
7									
8									
9									

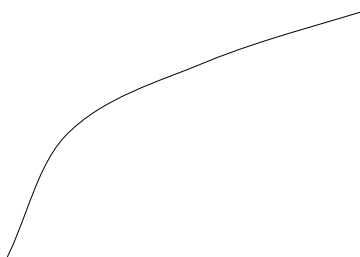
Timeline Chart

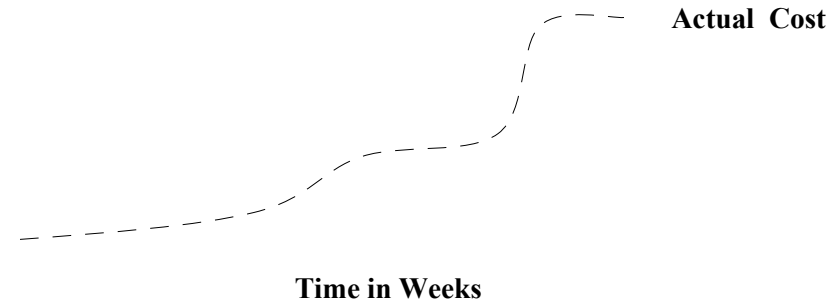
4.4 Cost Monitoring

- An important component of project control is cost monitoring.
- Cost monitoring provides an indication of the effort that has been given to the project.
- Sometimes, more cost is incurred to complete the activities to keep the project on schedule.
- A cumulative cost chart is depicted below:

Cumulative Cost

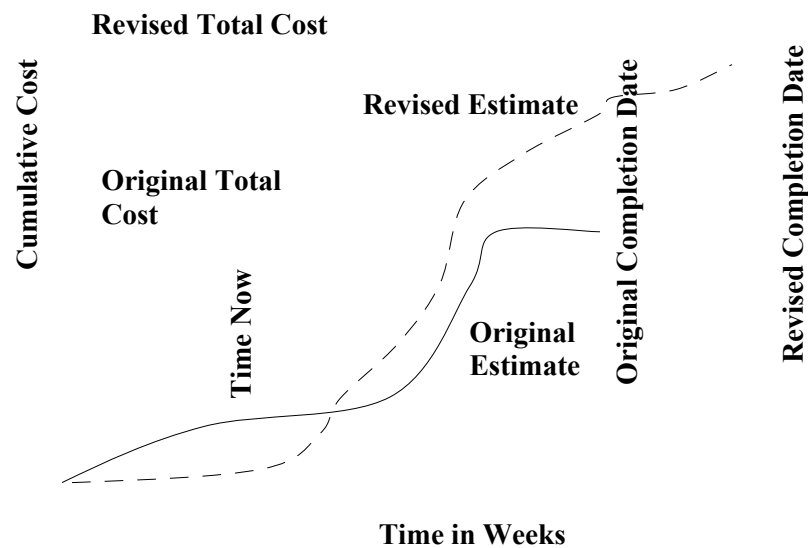
Planned Cost





Cumulative Cost Chart

- The chart does a comparison between the actual and the planned expenditure.
- These charts become more useful for estimating future costs.
- When revision of estimated cost and completion date are done, the same can also be expressed in the revised cumulative chart.



Revised Cumulative Cost Chart

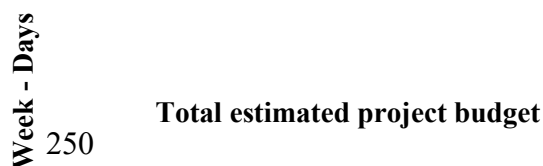
4.5 Earned Value Analysis

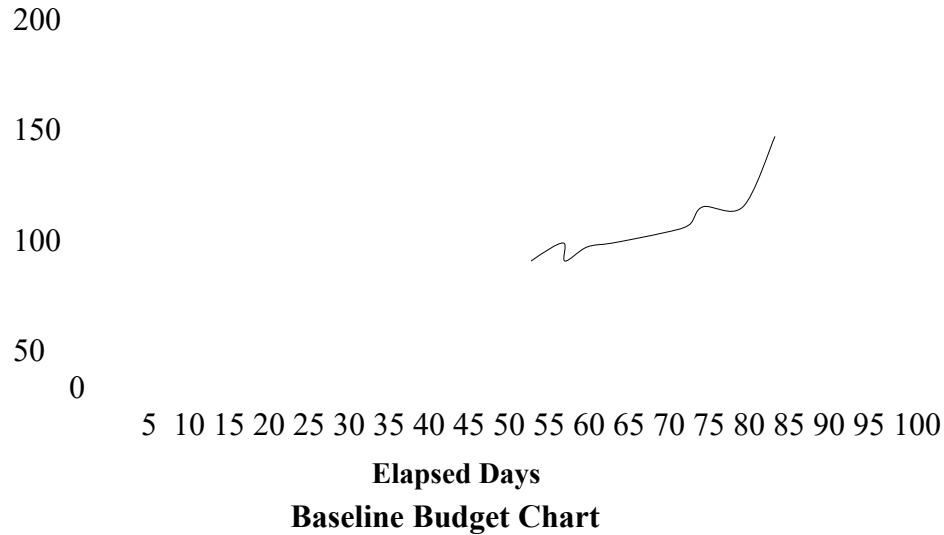
- An assigned value to each task or work package based on original cost forecasts yields earned value for the project.

- The assigned value is the original budgeted cost value and termed as a **planned value (PV)** or **budgeted cost of work schedule (BCWS)**.
- **Earned value(EV)**denotes the total value credited to a project at any point. It is also termed as **budgeted cost of work performed (BCWP)**.
- Common methods used in assigning an earned value are:
 - **0/100 technique:** Task value is assigned zero till completion and the budgeted value is 100%.
 - **50/50 technique:**Task value is assigned 50% and then increased to 100% once it completes.
 - **Milestone technique:**Task is assigned a value based on the achievement of milestones as part of original plan.
- Out of all these method, the 0/100 technique is used because the other techniques are not suitable for longer duration cost estimation.

4.5.1 Baseline Budgets

- To setup an earned value analysis, the first step is to create a baseline budget. The baseline budget shows the forecast growth of the project plan in earned value with respect to time.
- Common ways of measuring earned value in software development process is persons-hours or work-days.
- An 0/100 technique can be used to get the creditability of earned value.
- A typical baseline budget chart is given below and it depicts the scheduled completion of all activities involved in the development of the project

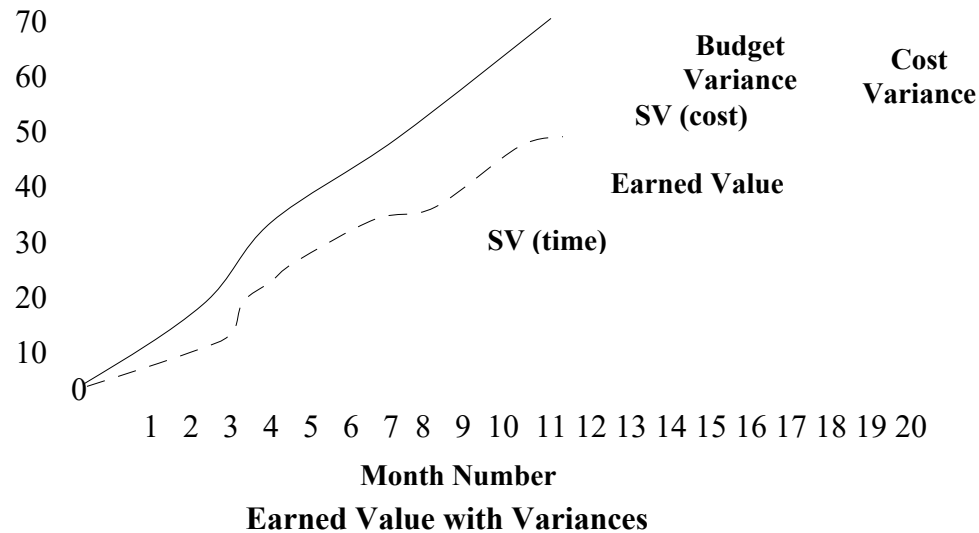




4.5.2 Monitoring Earned Value

- The next step in earned value analysis is to monitor the project progress.
- Monitoring process indicates the completion of tasks and includes the activity start and milestone achievement of the project.
- The **actual cost (AC)** is calculated by the actual cost of each task and is also called as **actual cost of work performed (ACWP)**.
- Certain inferences can be obtained from the earned chart such as:
 - **Schedule variance (SV)** : the difference between the earned value and the planned value indicates the degree of the completed work with the planned.
 - **Cost variance (CV)**: the difference between the earned value and the actual cost of a completed work results in cost variance. A positive CV value indicated that the project is under control and a negative CV denotes that the actual cost incurred is much more than the planned one.
- The diagram depicts the earned value analysis along with the schedule and cost variance.





4.5.3 Performance Ratios

- Performance ratios defines two index values namely **Cost Performance Index (CPI)** and **Schedule Performance Index (SPI)**.
- Cost performance index and Schedule performance index values are calculated by the formulas,

$$\text{CPI} = \text{Earned value} / \text{Actual Costs}$$

$$\text{SPI} = \text{Earned value} / \text{Planned value}$$

- When these indices refers to a greater value it means that the work is completed better than planned and if the value is less, it means that the work is more costlier than planned.

4.6 Project Tracking

4.6.1 Prioritizing Monitoring

The list of priorities defined in the level of monitoring are:

- **Critical path activities:** These denote those activities in the critical path that are delayed in project completion date.
- **Activities with no free float:** These delayed activities will have a delay in subsequent ones but still stick on target. These activities can have a serious effect on the resource schedule because the subsequent activities have to wait for its completion.

- **Activities with less than a specified float:** If there is a very little float in the activity say less than one week, these activities must be monitored very closely.
- **High risk activities:** These high risks are identified in the risk management plan itself and these results in over spending.
- **Activities using critical resources:** Critical activities are very expensive and are available only for a limited period and require high level of monitoring.

4.6.2 Getting back Project on Target

- Projects are subjected to delays and unexpected events.
- The project manager must ensure that the project scheduled end dates are unaffected at any circumstances.
- To maintain the project within the completed time, duration of some activity of the project can be delayed or shorten to fit into the time limit.
- The strategies involved in getting back the project to target are;
 - Critical path shortening
 - Reconsidering precedence requirements

Critical Path Shortening

- Delayed projects can often be brought back on track by shortening activity times on the critical path.
- Critical path is determined by the overall duration of the project.
- By increasing the resources for the critical path activities results in completion of the activity before time and the resources can be prolonged for a longer duration.
- At the same time, the resources used must be effectively allocated to all the activities so that no resources are idle at any point of time.
- Swapping of critical and non- critical activities can also be used to shorten the time limit and bring the project back to target.
- One disadvantage of shortening critical path is that, it produces many more paths while shortening which can become critical.

Reconsidering Precedence Requirements

- The project can be brought back to target by defining constraints to certain activities that affect the other activities for its completion.

- A precedence constraint activity can be sub-divided into a component that can start immediately.
- Altering these constraints would have a major impact on the quality factors, the risk involved, which can cause a delay in carrying out the activities.

4.7 Change Control

- Change control implies the authority to approve and rank the changes. It combines the automated tool with human to provide a mechanism for control of change.
- Any changes or alteration done to a single document often implies changes to other documents as well.
- Change request is evaluated to assess the technical aspect of configuration items and the budget.

4.7.1 Role of Chang Control Manager

- The responsibilities of change control manager or the configuration librarian are:
 - Identification of configuration items that are subjected to change control.
 - All project documentation and software products must be maintained in the central repository.
 - A formal set of procedures have to be setup to have control over changes.
 - Maintenance of library items in the repository

4.7.2 Change control Procedures

- Change control authority (CCA) makes the final decision on the status and the priority of the change based on the change report.
- Guidelines of a change control procedures includes:

Need for change is identified

Change requested by the development staff

Management considers the change request

Change control authority

Change needed, staff member is elected to handle

Change control denies

Prepares report based on cost and other factors

User is informed

Development staff reports user about change

User decides to accept the change or not

If user accepts, copies of master are chosen

User do not accept the change

Copies are modified, recompiled and tested

Management is informed

New version of the product is notified

Software released for user acceptance testing

If user is satisfied, master copies are replaced with change

New products are released after change

Change Control Procedure

4.7.3 System Scope Changes

- Any changes done leads to changes in the size of the system which gradually increases.
- The changes can be either from the management or from the user.

- For every change that is implemented, the scope of the developing project must be very carefully monitored and controlled.
- The changes made should not make the system to be inconsistent by affecting the estimating factors.

4.8 Software Configuration Management

Throughout development, software consists of a collection of items (such as programs, data and documents) that can easily be changed. During software development, the design, code, and even requirements are often changed, and the changes occur at any time during the development. This easily changeable nature of software and the fact that changes often take place require that changes be done in a controlled manner.

Software configuration management (SCM) is the discipline for systematically controlling the changes that take place during development. Software configuration management is a process independent of the development process largely because most development models cannot accommodate change at any time during development. SCM can be considered as having three major components:

- ✓ Software configuration identification
- ✓ Configuration control
- ✓ Status accounting and auditing

Configuration identification

The first requirement for any change management is to have clearly agreed-on basis for change. That is, when a change is done, it should be clear to what changes has been applied. This requires baselines to be established. A baseline change is the changing of the established baseline, which is controlled by SCM.

After baseline changes the state of the software is defined by the most recent baseline and the changes that were made. Some of the common baselines are functional or requirements baseline, design baseline, and product or system baseline. Functional or requirement baseline is generally the requirements document that specifies the functional requirements for the software. Design baseline consists of the different components in the software and their designs. Product or system baseline represents the developed system.

It should be clear that a baseline is established only after the product is relatively stable. Though the goal of SCM is to control the establishment and changes to these baselines, treating each baseline as a single unit for the purpose of change is undesirable, as the change may be limited to a very small portion of the baseline.

Configuration control

Most of the decisions regarding the change are generally taken by the configuration control board (CCB), which is a group of people responsible for configuration management, headed by the configuration manager. For smaller projects, the CCB might consist of just one person. A change is initiated by a change request.

The reason for change can be anything. However, the most common reasons are requirement changes, changes due to bugs, platform changes, and enhancement changes. The CR for change generally consists of three parts. The first part describes the change, reason for change, the SCIs that are affected, the priority of the change, etc.

The second part, filled by the CM, describes the decision taken by the CCB on this CR, the action the CM feels need to be done to implement this change and any other comments the CM may have. The third part is filled by the implementer, which later implements the change.

Status accounting and auditing

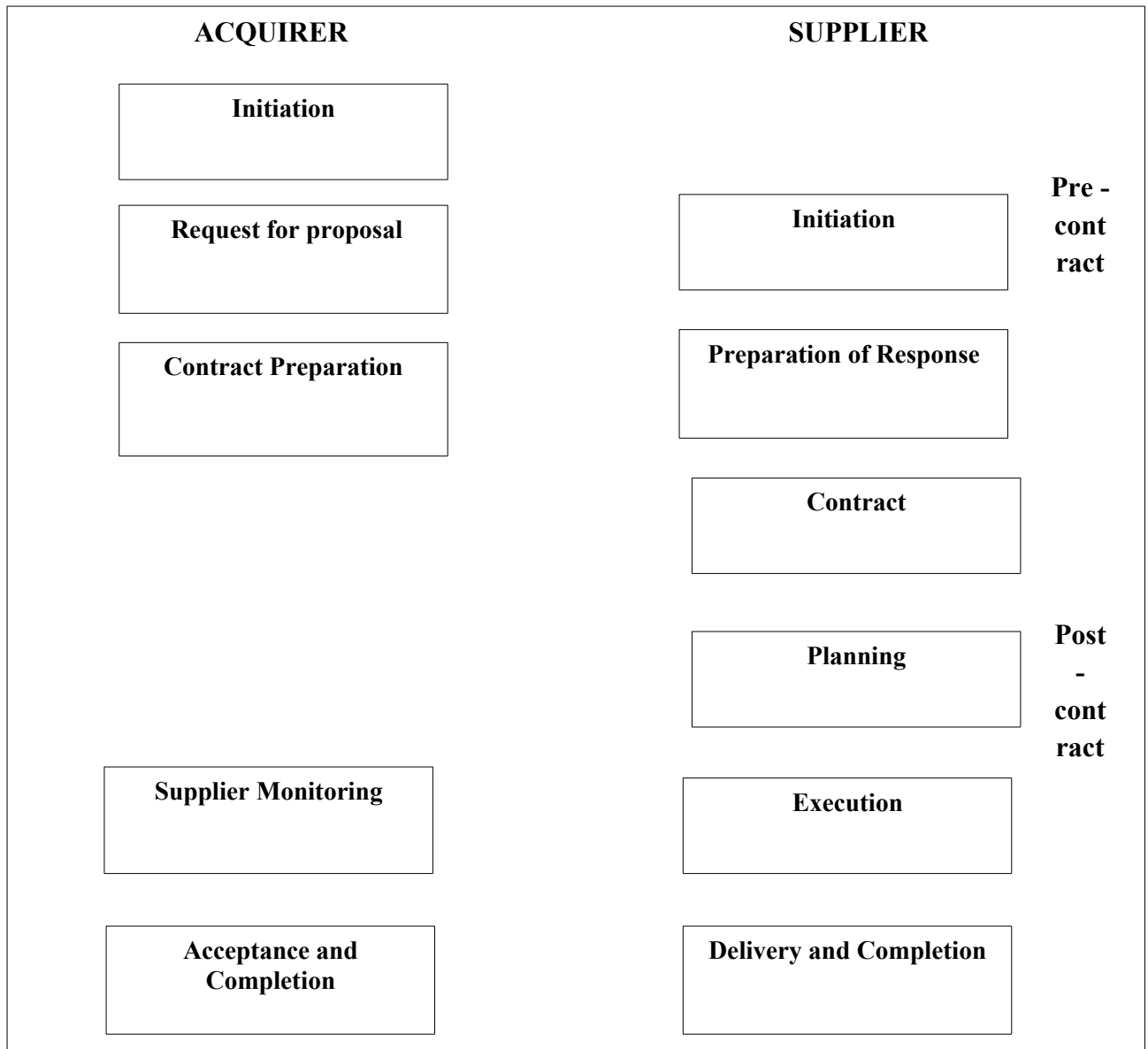
For status accounting, the main source of information is the CRs and FRs themselves. Generally, a field in the CR/FR is added that specifies its current status. The status could be active, complete, or not scheduled. Information about dates and efforts can also be added to the CR, the information from the CRs/FRs can be used to prepare a summary, which can be used by the project manager and the CCB to track all the changes.

4.9 Managing Contracts

4.9.1 Introduction

- The acquisition and supply process are depicted for pre-contract and post-contract as follows:
- The success of a contract requires considerable amount of time management.
- An ISO 12207 standard defined for acquisition and supply of software defines five major processes namely,
 - Acquisition

- Supply
- Operation
- Maintenance
- Development



ISO 12207 Acquisition and Supply process

- The initiation activity starts with the acquirer describing the preparation for the invitation to tender.

- System requirements are broader and are not related to software alone but depends on the changes in the organizational environment.
- Software requirements specifically relate to the software components within the delivered system and can be extracted from broader system requirements.
- Request for proposal of the project contains the system requirements, scope of the system, instruction for the bidders, list of software products, subcontractors detail and other technical constraints.
- The criteria for selecting the supplier will have to done very carefully by the acquirer done by joint reviews, verification and validation.
- As the supplier delivers, the acquirer conducts the review tests and is satisfied accepts the software and sign-off as completed.

4.9.2 Supply Process

- The supplier process activities will need to undertake in response to the request of supplier.

Initiation: The process is started when a request for a proposal from an acquirer and the supplier initiates the work.

Response Preparation: The response is prepared with expert knowledge drawn from various people.

Contract: Every activity is handled well, then the acquirer accepts the supplier and the details of the contract are negotiated and signed.

Planning: A detailed plan is developed of how the work has to be carried out.

Execution and Control: The detailed plan can be executed and the development process is invoked. The supplier must monitor and have control over the product quality in identifying, analyzing and providing resolutions.

Review and Evaluation: The acquirer reviews the progress of product information which are needed to be accessed.

Delivery and Completion: Post-delivery process has to be defined in view of the management plans.

4.9.3 Types of Contract

- Services are the external resources that are required for setting up a new system.
- **Based on the supply of a completed software package** the contracts can be classified as
 - **Bespoke system:** This kind of system is developed for an individual that is created from scratch.
 - **Off-the-shelf:** This package denotes what the user buys as it is and called as shrink-wrapped software.
 - **Customized off-the-shelf:** This system represents a basic core system that is modified based on the requirements of the client.
- **Based on how the payment is made to the supplier**, the contracts can be classified as,
 - **Fixed price contracts:** Here, the price is fixed when the contract is signed. There will be no changes in the contract terms and the payment must be made towards the end of the work.

Advantages of Fixed price contracts

- ❖ Customer expenditure is well-known.
- ❖ Motivation of the supplier towards delivering the product with cost-effective.

Disadvantages of Fixed price contracts

- ❖ Supplier absorbs the risk in original estimate and allows higher prices to allow contingency which is added as a margin quoted in the tender.
- ❖ Requirements once defined are difficult to modify which can cause friction between supplier and customer.
- ❖ Initially, the supplier will quote low price but as requirements are put forward, the supplier demands a higher price.
- ❖ Threat to the quality of the system can occur if the price is fixed.
- **Time and Materials contracts:** Here, the customer is charged with a fixed rate per unit of effort. This also estimates the overall cost based on the customer's requirements and it is not based on the final payment.

Advantages of Time and Materials contracts

- ❖ Changing requirements can be done very easily.
- ❖ The customer is not worried about the price pressure.

Disadvantages of Time and Materials contracts

- ❖ Customer cannot include all the requirements needed by them.
- ❖ The supplier will not like to work in a system where the scope defined is out of control.
- **Fixed price per unit delivered contracts:** This type of contract is based on function point counting. Along with the size of the system which includes LOC, a price per unit is also quoted. In this system, the scope grows during the development process.

Advantages of Fixed price per unit delivered contracts

- ❖ Unlike the time contract, the customer can understand the changing requirements
- ❖ Pricing schedules can be compared.
- ❖ Supplier does not get affected by risk increasing functionality.
- ❖ Supplier has efficiency to deliver the cost-effective manner.
- ❖ Development contract includes both the analysis and design stages of the project.

Disadvantages of Fixed price per unit delivered contracts

- ❖ It is very difficult to measure the software size.
- ❖ The requirement changes sometimes affect the transactions that are not included in the function point count.

➤ **Based on the approach used in contractor selection** the contracts can be classified as

- **Open tendering process:** The request for proposal must be considered and evaluated with the original conditions. Every supplier can bid to supply the goods and services. The evaluation process can be time-consuming and also expensive in open tendering process.
- **Restricted tendering process:** Here, bids can be made only by suppliers who have been invited by the customer. This is a better approach than the open tendering but has some risk factors.
- **Negotiated procedure:** In particular instances, the restricted tendering process fails because of the defects which lead to additional payment towards the completion of the project.

4.9.4 Stages in Contract Placement

- The main stages in contract placement are given below:
 - **Requirement analysis**
 - **Evaluation plan**
 - **Invitation to tender**
 - **Evaluation of proposals**

Requirement analysis

- Preparation of an requirement document containing the following:
 - Introduction
 - Description of the existing system
 - Current environment of the system
 - Customer's future plans
 - System requirements based on either mandatory or desirable
 - Deadlines have to be defined
 - Additional information requires from the potential suppliers

Evaluation plan

- The evaluation plan contains:
 - Preparing a plan to evaluate the submitted proposals.
 - Checking for the mandatory requirements that have been defined to meet the objectives.
 - Evaluating the desirable requirements
 - Validating the quality of the software system
 - Cost incurred for the lifetime of the proposed system

Invitation to tender

- This is also termed as request for proposal and contains:
 - System requirements
 - Defining the scope of the system
 - Instruction to the bidders
 - List of the software products
 - Control of the subcontractors resulting in MoA
 - Technical constraints

Evaluation of proposals

- This evaluation has to be done in a planned manner. The process of evaluation includes:
 - Scrutiny of the proposal documents
 - Questioning supplier representatives
 - Giving demonstrations
 - Visiting the site of the development process
 - Conducting practical tests.

4.9.5 Typical Terms of Contract

- The contents of a typical terms of contract are listed below:
 - **Definitions:** The exact meaning of who is a supplier and who is a customer.
 - **Forms of agreement:** Categorizing whether the contract is sale, a lease, or a license.
 - **Goods and services to be supplied:** This contains the actual list of individual pieces of equipment that has to be delivered with specific model numbers. The services includes proper training, documentation, installation, conversion of existing files, maintenance of agreements and insurance related issues.
 - **Ownership of the software:** There are two possible ownership that can exists; one with the customer and the other with the supplier. Supplier provides a license to the user to use but that does not mean the ownership changes. Any assignment of copyright must be in writing.
 - **Environment:** The basic working environment facilities have to be provided by the supplier and the customer such as electricity supply.
 - **Customer commitments:** Customers have to provide with the basic accommodation facilities even though the work is carried out by external contractors.
 - **Acceptance procedures:** Various tests are conducted and the system is accepted after the procedure for signing off the testing process is completed.
 - **Standards:** Every product that is supplied must abide by the standards relating to its development and its documentation.

- **Project and quality management:** The quality that is expected by the management for the project can be influenced by conducting review meetings and obtaining the progress information of the project.
- **Timetable:** A schedule is drawn to describe the different tasks and activities that has to be carried out during the development process.
- **Price and payment method:** Payment must be made based on the price that has been defined in the agreement ensuring that the goods and services are satisfactory.
- **Miscellaneous legal requirements:** A contract must be defined within the legal jurisdiction stating the liabilities that are applied to sub contractors involved in the process. Liquidated damages can cause financial losses where the customer suffers if the supplier is not able to oblige.

4.10 Contract Management

- Contract management studies and monitors the conversation between the supplier and the customer while the contracted work is being carried out.
- Customer can make changes to the future direction of the project and make decisions.
- The entire project will require representative of the supplier and the customer to interact with each other at different points in the development process.
- Activities involved in contract management include:
 - Identifying customer approval;
 - Negotiating successfully;
 - Project deliverables;
 - Managing change;
 - Decision making;
 - Legal obligations;
 - Business laws.

Accepting the Contract

- Customer has to undergo acceptance testing towards the end of the process.
- Every contract would have defined a time limit for the acceptance testing and the result has to be produced before the time expires.

- Certain software suppliers are concerned with pre-acceptance testing where the user tests the system than the developer.
- The supplier will not like to retain their staff to a specific project after its completion.
- Customer finds that the modifications needed by them are handled only by the junior level staffs that are not aware of the delivered product.
- All the payment to the supplier solely depends on the acceptance testing.
- Every bug that is raised must be fixed within the period of warranty.

UNIT V –STAFFING IN SOFTWARE PROJECTS

5.1 Managing People

Understanding Behavior

- Handling of projects with practical experience becomes a vital role in the aspect of project management.
- The managers must be able to decide on whether it is better to have experienced staff or get an expert advice.
- There are numerous theories defined to explain people's behavior.
- The theories are structures based on "If A is the situation then B is likely to be the solution".
- Other than the structures, there a wide range of influences on a situation which are invisible to the users which makes it difficult to decide on the solution.

5.2 Organizational Behavior

5.2.1 Work Objectives

- Fredrick Taylor analyzed the productive way of doing things and trained the workers with these objectives:
 - **To select the best people for the job**
 - **To instruct them in the best methods**
 - **To give them incentives based on their performance**
- These work objectives defined by Taylor emphasis exclusively on the financial basis of the staff motivation and performance-related pay.
- This encouragement to the staff will help the project group to work together in achieving their goals which ultimately increases the productivity.
- People may be motivated by money, but they are motivated by other factors as well.

5.2.2 Theory X and Theory Y

- Some managers' work for money being instrumental or called as cash-oriented persons can be categorized based on their individual attitudes.

- Donald McGregor labeled two different attitudes as Theory X and Theory Y.
- **Theory X** includes
 - On an average, not every human likes to work.
 - Somebody must have the control and direct the person to work.
 - Generally, people do not like hold responsibilities.
- **Theory Y** includes
 - People must like to work not forced to do it.
 - External control is not the way to reach organizational goals.
 - There must a commitment towards the work allocated to individuals.
 - An average human can learn to accept and seek responsibility.
 - Creative qualities must be widely distributed.
- Individual's behavior towards the organization can be observed when their boss is not available.
- Theory X environment makes everybody to relax which can be seen visibly whereas Theory Y is more a goal oriented approach of the people involved in the development.
- A reward does not need to be a financial reward but it could be something like a sense of achievement.
- This theory explains the expectations which have a greater influence towards the organizational behavior.

5.3 Best methods for staff selection

5.3.1 Selecting the right person for the right job

- Taylor formulated this factor of selecting the right person for the right job.
- The other factors which includes the use of software tools, methodologies, programming productivity and so on.
- It is always better to have the best people employed in the right place of work for effective productivity.
- Say for example, a study on comparing experienced programmer in debugging a code with a less experienced person was futile when the results

were drawn.

- According to Gerald Weinberg, “Most programmers prefer to work alone where they are not disturbed by other people”.

5.3.2 Recruitment Process

- There is a lot of stress for every project manager about choosing the right people to make up their team.
- Recruitment is an organizational responsibility process of selecting the person from their organization.
- Meredith Belbin categorizes people in recruiting process into two different types.
 - **Eligible candidates** are those persons who have the right information needed by the organization in paper, i.e. the curriculum vitae contains the right number of years and right paper qualifications.
 - **Suitable candidates** can actually do the job well but are not officially eligible. Ideal candidates once given a post are likely to be more loyal towards the welfare of the organization.
- Actual skills have to be taken into account while selecting a person rather than mere eligibility.
- The recruitment policy must avoid discrimination of race, gender, age or irrelevant disabilities.
- A recruitment process must include:
 - **Creation of job specification:** type of task that has to be carried out must be documented and agreed.
 - **Creation of job profile:** constructs a profile of the person needed to carry out the job with qualities, qualifications, education and experience.
 - **Obtaining applicants:** placing an advertisement within the organization or the local press to get a maximum number of potential applicants.
 - **Examine Curriculum Vitae:** all the received CV's are compared with the job holder profile and if satisfied are called for interview.

- **Interviews:** these include the aptitude tests, technical tests, personality tests and examination of previous work. Group discussions are also used for evaluating and examining the statements provided in the CV.
- **References:** need to be verified and a medical examination test can also be done if needed.

5.3.3 Instruction in the Best Methods

- The responsibility of the team leader becomes very high in case of recruiting new members into the team.
- Half way through the project process, maximum effort must be taken to make the induction of a new member into the team as an effective one.
- Team members must be continually assessed by the team leader to meet their demands in the development process.
- Proper training must be given to every staff member involved in the development of the project.
- There are many companies which provide training to the staffs by conducting specific courses and giving hand-on training about the new software tool by demonstration.
- Whatever may be, the training process must be actually implemented by the team members as a whole in order to meet the objectives.

5.4 Motivation

Motivation Theories

There are various theories formulated by different persons for motivating the people to work. They are,

- Taylorist Model
- Maslow's Hierarchy of Needs
- Herzberg's Two-Factor Theory
- Expectancy Theory of Motivation.

5.4.1 Taylorist Model

- In this model, Taylor emphasis on the piece-rates and day-rates.
- Piece-rates are those where the workers are paid a fixed sum for each single item they produce whereas day-rates refer to the daily pay that is given to the workers on a timely basis.
- The tendency towards dispersed or virtual projects where the staffs either work in organization or work at home has a difference in the payment based on time worked.
- The amount paid to the workers will not directly relate to maximize the output in order to maximize their income.
- The amount of output will normally depend on the working group and not based on an individual.
- A reward based on piece-rates is directly proportional to the work produced. But a support team cannot be adjudged by a single person, instead it is group activity and the reward must be given to the group as a whole.
- In Taylorist model, the reward system makes excessive distinctions between co-workers that result in damaging morale and productivity.
- This can be balanced by giving bonuses to project team members after completion of a successful project.

5.4.2 Maslow's Hierarchy of Needs

The basic human needs placed by Maslow in an ascending order of importance are:

1. **Physiological Needs:** These are the basic needs for sustaining human life itself, such as food, water, warmth, shelter, and sleep. Maslow felt that until these needs are satisfied to the degree necessary to maintain life, other needs will not motivate people.
2. **Security or Safety Needs:** These are the needs to be free of physical danger and of the fear of losing a job property, food, or shelter.

3. **Affiliation or Social Needs:** Since people are social beings; they need to belong, to be accepted by others. It includes friendship, the need to love and be loved, socializing, etc.

4. **Esteem Needs:** Once people begin to satisfy their need to belong; they tend to want to

be held in esteem both by themselves and by others. This kind of need produces such satisfactions as respect, power, prestige, status, and self-confidence.

5. **Self-actualization Needs:** This is the highest need in the hierarchy. It is the desire to become what one is capable of becoming—to fully realize one's potential and to accomplish what one is capable of achieving.

5.4.3 Herzberg's Two-Factor Theory

- People tend to be dissatisfied about their job due to certain factors. They are
 - **Hygiene or maintenance factors**
 - **Motivators**
- A **hygiene factor** makes the person dissatisfied if they are not rightly used. For example, the working condition of the worker.
- **Motivators** can make the person feel that the job is worth doing it, like a sense of achievement or the challenge of the work.
- Higher-level of maintenance factors can be provided by large organizations whereas better motivation can be provided to workers who work in smaller organizations.

5.4.4 Expectancy Theory of Motivation

- Vroom identified three influencing factors on motivation. They are:
 - **Expectancy**
 - **Instrumentality**
 - **Perceived value**
- **Expectancy** is a belief that working harder will lead to better performance.

- **Instrumentality** is the belief that better performance will be rewarded.
- **Perceived value** denotes the resulting reward.
- When all these factors are high, then the motivation level will also be high. At the same time, a zero level for any one of the factor can remove motivation completely.
- For example, when the developer is suppose to get a software package supplied by a third party to work and it contains a bug, the worker gives up since how much hard work the worker puts in it will not lead to success denotes **zero expectancy**.
- On the other side, if the user is not using the package supplied by the developer, instead the user works on an alternative tool, it makes the developer feel that it is waste of time and leads to **zero instrumentality**.
- Suppose if the user is using the package but keeps on complaining about the package and makes the developer responsibility for all short-comings, then at some point of time the developer will not like to get involved for implementing a newer package which leads to **low perceive value of reward**.

5.5 The Oldham-Hackman job characteristic model

- Oldman and Hackman coined a rule that managers should group together the elements of tasks that is carried out must be meaningful and satisfying assignments.
- The satisfaction of any job will depend on the following factors:
 - **Skill variety**
 - **Task identity**
 - **Task significance**
 - **Autonomy**
 - **Feedback**

5.5.1 Elements of Tasks

- Factors that make the job meaningful to the person who is doing it are skill

variety, task identity and task significance.

- **Skill variety** is the number of different skills that the job holder has the opportunity to exercise.
- **Task identity** is the degree to which the person's work and its results are identifiable as belonging to the person.
- **Task significance** is the degree to which the job has an influence on others.
- The **autonomy** factor is the discretion about the way the person works.
- **Feedback** is the information the person receives back about the results of his/her work.
- Personal growth needs and their working environment also influence the perception of the job.
- In general, activities of the developing product should be designed in such a manner that the person must feel personally associated with it.

5.5.2 Methods to Improve Motivation

- Managers must adopt the factors listed below to improve motivation.
 - **Set specific goals**
 - **Provide feedback**
 - **Consider job design**
- **Specific goals** must be demanding goals but yet acceptable by staff and should gain approval from them.
- **Providing feedback** reflects the performance of the staff about how they are progressing.
- **Considering job design** makes the job more interesting and provides the staff with greater responsibility.

5.5.3 Measures for enhancing Job

- Managers must involve the following measures to enhance the job design: .
 - **Job enlargement**
 - **Job enrichment**
- **Job enlargement** is exactly reverse of specialization where the person doing

the job carries out a wider variety of activities.

- Say for example, a software developer associated with maintenance group might be given additional responsibility for specifying minor changes in other phases.
- **Job enrichment** is where the job holder carried out tasks at managerial and supervisory level.
- Say for example, programmers in maintenance group might be given authority to accept requests for changes for a very small period (five days) without getting manager's approval.

5.6 Ethical and Programmed Concerns

Ethics relates to the moral obligation to respect the rights and interests of others – goes beyond strictly legal responsibilities.

Three groups of responsibilities:

- Responsibilities that everyone has
- Responsibilities that people in organizations have
- Responsibilities relating to your profession or calling

Organizational ethics

There are some who argue that ethical organizational ethics are limited:

Stockholder theory (e.g. Milton Friedman). An employee's duty is to the owners of the business (which often means the stakeholders) above all others – although legal requirements must be met.

Competitive relationships between businesses.

Competition may cause you to do things that could have a negative impact on the

owners or employees of competitive businesses Uniform Treatment

One example of organizational ethics is the uniform treatment of all employees. Small business owners should treat all employees with the same respect, regardless of their race, religion, cultures or lifestyles. Everyone should also have equal chances for promotions. One way to promote uniform treatment in organizations is through sensitivity training. Some companies hold one-day seminars on various discrimination issues. They then invite outside experts in to discuss these topics. Similarly, small company managers must also avoid favoring one employee over others. This practice may also lead to lawsuits from disgruntled employees. It is also counterproductive.

Social Responsibility

Small companies also have an obligation to protect the community. For example, the owner of a small chemical company needs to communicate certain dangers to the community when explosions or other disasters occur. The owner must also maintain certain safety standards for protecting nearby residents from leaks that affect the water or air quality. There are state and federal laws that protect people from unethical environmental practices. Business owners who violate these laws may face stiff penalties. They may also be shut down.

Financial Ethics

Business owners must run clean operations with respect to finances, investing and expanding their companies. For example, organizations must not bribe state legislators for tax credits or special privileges. Insider trading is also prohibited. Insider trading is when managers or executives illegally apprise investors or outside parties of privileged information affecting publicly traded stocks, according to the Securities and Exchange Commission. The information helps some investors achieve greater returns on their investments at the expense of others. Executives in small companies must strive to help all shareholders earn better returns on their money. They must also avoid collusive arrangements with other companies to deliberately harm other competitors.

Considerations

A small company's organizational ethics can also include taking care of employees with mental illnesses or substance abuse problems, such as drug and alcohol dependency. Ethical business owners help their employees overcome these types of problems when possible. They often put them through employee advisor programs, which involves getting them the treatment they need. Employees may have issues that lead to these types of problems. Therefore, they deserve a chance to explain their situations and get the help they need.

Professional ethics

Professionals have knowledge about the technical domain that the general public does not. Ethical duty of the expert to warn lay people of the risks involved in a particular course of action. Many professions, or would be professions, have codes of conduct for their members

5.7 Working in teams

- Not all people involved in the development process like to work in groups.
- But major software projects always have to work in groups and many people do not like to work in groups.
- Any organization involved in the development process will have various departments reflecting its structure.
- Formal groups can be formulated based on the different departments and task groups can be formed based on specific tasks carried.
- Task group can contain different people from different departments to work together to carry out a specific task.
- Every task group formed for specific activities to be carried out are dissolved once the task is completely achieved.

5.7.1 Becoming a team

- Making people work together is the most difficult task that the project manager has to carefully handle.
- A team cannot perform instantly; it has to develop over time.

5.7.2 Team Formation Model

- Every team has to go through five different stages of development as depicted in the Team Formation Model namely,
 - **Forming**
 - **Storming**
 - **Norming**
 - **Performing**
 - **Adjourning**
- **Forming** : basic ground rules and general behavior are set up to try and get to know each other in the team.
- **Storming**: grouping methods of operation have to establish as there is a chance of conflicts arising due to leadership.
- **Norming**: a group identity emerges as the conflicts are largely settled.
- **Performing**: how the tasks are handled by the team.
- **Adjourning**: disbanding of the group.

5.7.3 Individual Characteristics

- Any project team must be formed with the best mix of different personalities.
- Belbin formulated the need of balanced teams based on individual characteristics of people.
 - **Chair**: these people must be good at conducting meetings, must be calm, strong and tolerant. Need not be excellent leaders.
 - **Plant**: these people must be good at generating ideas and giving potential solution to problems.
 - **Monitor-Evaluator**: they must be good evaluators and best in selecting

the most feasible solution.

- **Shaper:** helps in directing the team's attention to important issues.
 - **Team worker:** must be efficient in creating a good working environment.
 - **Resource investigator:** helps in finding resources in terms of both physical resources and information.
 - **Complete-finisher:** people who are concerned with completing the tasks.
 - **Company-worker:** must be good team player willing to undertake less attractive work for team's success.
- To be a good team member, the person must be flexible, restrained, timely and keeping the common goals of the team in mind all the time.

5.7.4 Group Performance

- There is a strong question raised often: **"Are groups more effective than individuals working alone?"**.
- It is the responsibility of the project manager to distinguish the tasks which are supposed to be carried out together and those tasks to be carried out by individuals.
- Some works yield better results when worked together as a team, while some others are slowed down because of the work be compartmentalized based on individuals.
- There are four different ways of categorizing group tasks. They are:
- **Additive Tasks:** in this the effort of every person are added to reach the final result. People involved in additive tasks are interchangeable.
 - **Compensatory Tasks:** here, the judgements of individual group members are taken and the results are then averaged. These result in effective group work rather than the efforts of individuals.
 - **Disjunctive Tasks:** these tasks have only one correct solution to the task. Here, if someone comes with a solution and everybody in the

team accepts it.

- **Conjunctive Tasks:** here, the progress is governed by the rate of the slowest performer where until each and every person completes their own tasks, the overall tasks do not attain its completion. In this case, a cooperative attitude of the team becomes productive.
- A major problem can arise with additive tasks that lead to **social loafing**.
- **Social loafing** is a problem where some individuals do not make their proper contribution when carrying out group assignments.

5.8 Decision making

5.8.1 Categories of Decisions

- Decision making process can be categorized into **structured** and **unstructured**.
 - **Structured** decisions are generally simple where the rules are applied in a straight-forwards way.
 - **Unstructured** decisions are often more complex and require a great degree of creativity.
- The **amount of risk** and the **uncertainty** involved in the development process can affect the decision making process.

5.8.2 Obstacles to Good Decision Making

- Few factors that affect good decision making process are:
 - **Faulty heuristics:** the rules of thumb or heuristics are useful but can be misleading. These are based on mere stereotypes.
 - **Escalation of commitment:** this happens when a wrong decision is made which cannot be altered easily later.
 - **Information overload:** too much of information also might lead the decision process to choose a wrong one.

5.8.3 Group Decision Making

- In group discussions, different specialists and point of view of stakeholders

can be brought together to make a better decision.

- Decisions made by a team can be approved and accepted easily than decisions imposed by individuals.
- Every group meeting takes the collective responsibility of having properly briefed of solving complex problems.
- A group can arrive at better solutions for complex problems because the members of the group have complementary skills and expertise.
- Group meetings provide an opportunity for people to communicate freely and easily among the members of the group.
- Often, groups are less effective for poorly structured problems where brainstorming techniques can be used to help the groups to make it structured
- Even though, group decision making is effective in achieving solutions, it has been proved by research that people come up with more ideas individually than in groups.

5.8.4 Obstacles to Good Group Decision Making

- Group decision making process has its own disadvantages:
 - It is time consuming process.
 - Conflicts can arise among the members of the group.
 - Decisions can be influenced by dominant personalities.
- Once established the group norms can survive many changes of membership in the group.
- Experimental results have shown that people can modify their personal judgements to conform to group norms.
- Sometimes people in groups take hasty decisions that can cause more risk than when they make their decisions on their own known as risky shift.

5.8.5 Measures to reduce obstacles of Group Decision Making

- To make group decision making process to be more effective and efficient the **Delphi Technique** can be adopted.

- Delphi technique endeavours to collate the judgements of a number of experts without actually bringing them face to face.
- The set of procedures is carried out as follows:
 - Cooperation of a number of experts is enlisted.
 - Problem is presented to the experts.
 - Experts record their recommendations.
 - Recommendations are collated and are reproduced.
 - Collected responses are recirculated.
 - Experts comment on the ideas of others and modify their recommendations.
 - If the leader finds any discrepancy, the process is stopped, otherwise it is recirculated to the experts.
- This method can be adopted to geographically dispersed experts but the process becomes time consuming.

5.9 Team Structures

5.9.1 Department Structure

- Departmentalization of organizations depends on staff specialties, product lines, categories of customer or the geographical location.
- In general, the software development process approach is referred as either **function-oriented** or **task-oriented**.
- Function oriented approach deals with different groups that are formed based on their functional specialization.
- This approach leads to a more effective usage of staff both technically and in employing the standards that are needed to be concerned.
- With the task oriented approach, members are grouped together with respect to a specific task.
- The specific task has to be achieved by the group and the group is dissolved after the successful completion of the particular task.
- Departmentalization is also based on **life-cycle phase**.

- In project life cycle phases there are separate teams for development and maintenance.
- **Matrix** form of departmentalization can also be formed where there are two managers namely project manager and programming manager. The project manager deals with the day-to-day activities while the programming manager focuses on future career development.
- **Egoless programming** suggests that the programmers and the programming team leaders should read other people's programs so that the programs become a common property to both.

5.9.2 Team Structure

Team structure addresses the issue of organization of the individual project teams. There are mainly three formal team structures:

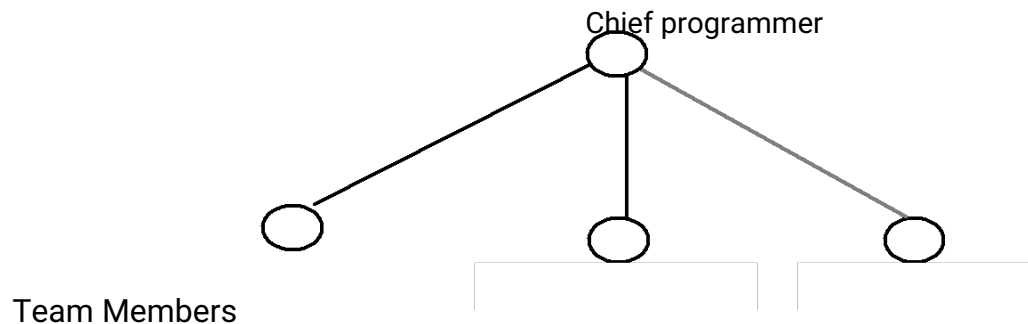
Chief programmer,
Democratic, and
The mixed control team organizations

Chief Programmer Teams

- If the number of groups is larger, then the work will be slower because of increased communication. So large projects must be formalized and must be represented in an centralized structure.
- One way to avoid this, to reduce the number of people and giving them more support to make the work done which led to the formulation of **chief programmer team**.
- The **chief programmer** defines the specification, design, code, tests and documents the entire software.
- The chief programmer can have a **co-pilot** who can assist in writing some code and discussions.
- An **editor** can be used to write up the documentation drafted by the chief programmer, along with a **program clerk** who maintains the actual code and a

tester who validates the code.

- The disadvantage of chief programmer teams is that the chief programmer is overloaded with lots of information and cannot manage at some point of time.
- **Extreme programming** concept can overcome this disadvantage where the software is developed by pairs of developers with a **chief programmer / co-pilot relationship**.



In this team organization, a senior engineer provides the technical leadership and is designated as the chief programmer.

The chief programmer partitions the task into small activities and assigns them to the team members.

He also verifies and integrates the products developed by different team members.

Advantages

The chief programmer provides an authority, and this structure is arguably more efficient than the democratic team for well-understood problems.

However, the chief programmer team leads to lower team morale, since team-members work under the constant supervision of the chief programmer.

This also inhibits collective and their original thinking.

The chief programmer team is subject to single point failure since too much responsibility and authority is assigned to the chief programmer. Since the chief programmer carries out many tasks individually, there is a danger of information overload on the chief programmer

Democratic Team Structure

The democratic team structure, as the name implies, does not enforce any formal team hierarchy. Decisions are taken based on discussions, where any member is free to discuss with any other matters. Typically, a manager provides the administrative leadership. At different times, different members of the group provide technical leadership.

Advantages:

The democratic organization leads to higher morale and job satisfaction.

Democratic team structure is appropriate for less understood problems, since a group of engineers can invent better solutions than a single individual as in a chief programmer team.

A democratic team structure is suitable for projects requiring less than five or six engineers and for research-oriented projects. For large sized projects, a pure democratic organization tends to become chaotic.

The democratic team organization encourages egoless programming as

programmers can share and review one another's work.

Disadvantages:

Consequently, it suffers from less man-power turnover

Mixed Control Team Structure

The mixed team organization, as the name implies, draws upon the ideas

from both the democratic organization and the chief-programmer organization. This team organization incorporates both hierarchical reporting and democratic set up.

The democratic connections are shown as dashed lines and the reporting structure is shown using solid arrows.

The mixed control team organization is suitable for large team sizes.

The democratic arrangement at the senior engineer's level is used to decompose the problem into small parts. Each democratic setup at the programmer level attempts solution to a single part. Thus, this team organization is eminently suited to handle large and complex programs.

This team structure is extremely popular and is being used in many software development companies.

5.10 Virtual Teams

A Virtual Team – also known as a Geographically Dispersed Team (GDT) – is a group of individuals who work across time, space, and organizational boundaries with links strengthened by webs of communication technology. They have complementary skills and are committed to a common purpose, have interdependent performance goals, and share an approach to work for which they hold themselves mutually accountable. Geographically dispersed teams allow organizations to hire and retain the best people regardless of location. A virtual team does not always mean teleworkers. Teleworkers are defined as individuals who work from home. Many virtual teams in today's organizations consist of employees both working at home and small groups in the office but in different geographic locations.

Why Virtual Teams?

Best employees may be located anywhere in the world.

Workers demand personal flexibility.

Workers demand increasing technological sophistication.

A flexible organization is more competitive and responsive to the marketplace.

Workers tend to be more productive – less commuting and travel time.

The increasing globalization of trade and corporate activity.

The global workday is 24 vs. 8 hours.

The emergence of environments which require inter-organizational cooperation as well as competition.

Changes in workers' expectations of organizational participation.

A continued shift from production to service/knowledge work environments.

Increasing horizontal organization structures characterized by structurally and geographically distributed human resources.

5.11 Communication genres

A major influence on the nature of communication genres is the constraints of time and place. Modes of communication can be categorized as combinations of two opposite: Same time/Different time and Same Place/Different Place.

	Same Place	Different Place
Same Time	Meetings, Interviews	Telephone, Instant Messaging
Different Time	Notice Boards	E-mail, Voice mail, Documents

The nature of the information to be conveyed:

- o What is the extent and complexity of the information to be conveyed?

A phone conversation if message is simple

- o Is it easy to understand? Is the context well known to both the sender and the recipient?

Two way communication

- o Where the communication is personally sensitive

Face-to-face contacts

At different stages of a project – different communication genres will be preferred

Early stages – meeting(s)

Team members need to build up their trust and confidence in their co-workers

Decision making

Intermediate stages (design) – teleconferencing

Activities executed in parallel

Some points needs to be clarified

Implementation stages - emails

Everyone knows his role, work can progress

5.12 Communication plans

A communications plan, in project management, is a policy-driven approach to providing stakeholders with information about a project. The plan formally defines who should be given specific information, when that information should be delivered and what communication channels will be used to deliver the information.

An effective communications management plan anticipates what information will need to be communicated to specific audience segments. The plan should also address who has the authority to communicate confidential or sensitive information and how information should be disseminated (email, web sites, printed reports, and/or presentations). Finally, the plan should define what communication channels stakeholders should use to provide feedback and how communication documentation will be archived as part of the project records.

In some organizations the communications management plan may also include a glossary of common project terminology that will be used within the project. This

glossary may define and include samples of templates, reports and forms that the project manager will use to communicate information.

The result of the communication process could be documented in a table with the following column headings.

- What – This contains the name of a particular communication event
- Who/target – The target audience for the communication
- Purpose – What the communication is to achieve
- When/frequency – If the communication by means of a single event, then a specific date can be supplied. If the event is a recurring one, such as a progress meeting, then the frequency should be indicated
- Type/method – The nature of the communication, for example a meeting or a distributed document
- Responsibility – The person who initiates the communication

5.13 Leadership

- Leadership means the ability to influence others in a group to act in a particular way to achieve group goals.
- A leader need not be a very good manager or vice-versa since managers have different roles such as organizing, planning and controlling.
- It is very difficult to list the common characteristics of good leaders.
- But every leader have a greater need for power and achievement and must have more self-control and self-confidence than others.
- Leadership is generally based on the idea of authority or power.

5.13.1 Positional Leadership Power

- Power can take the form based on the position of the person. Positional power can be analyzed as:
 - **Coercive power:** ability to force someone to do something by threatening punishment.
 - **Connection power:** based on having access to those who have power.

- **Legitimate power:** based on person's title giving a special status.
- **Reward power:** here the holder gives rewards to those who carry out tasks to the satisfaction of their leader.

5.13.2 Personal Leadership Power

- Personal power depicts the person's individual qualities. Personal power can be analyzed as:
 - **Expert power:** person who is capable of doing specialized tasks.
 - **Information power:** here, the holder has exclusive access to information.
 - **Referent power:** based on the personal attractiveness of the leader.

5.13.3 Leadership Styles

- In order to make best use of the expertise and commitment of the people involved the leaders must be an authoritative but at the same time more flexible and tolerant.
- Sometimes, the leaders must be democratic as well, to have a very disciplined execution of the plan.
- Leadership styles can be classified as:
 - **Directive vs. permissive**
 - **Autocratic vs. democratic.**
- **Directive autocrat** makes decisions alone and will be person very closely associated with the implementation.
- **Permissive autocrat** also makes decisions alone, but subordinates have latitude in implementation.
- **Directive democrat** makes decisions participative and will be a person very closely associated with the implementation.
- **Permissive democrat** also makes decisions participative, but subordinates have latitude in implementation.
- The emphasis is that there are no one best style of leadership which has to be chosen by the management but it truly depends on the situation.